Numerische Mathematik manuscript No.
(will be inserted by the editor)

Implicit Standard Jacobi Gives High Relative Accuracy

Froilán M. Dopico · Plamen Koev · Juan M. Molera

Received: date / Accepted: date; Version of May 4, 2009

Abstract We prove that the Jacobi algorithm applied implicitly on a decomposition $A = XDX^T$ of the symmetric matrix A, where D is diagonal, and X is well conditioned, computes all eigenvalues of A to high relative accuracy. The relative error in every eigenvalue is bounded by $O(\varepsilon\kappa(X))$, where ε is the machine precision and $\kappa(X) \equiv \|X\|_2 \cdot \|X^{-1}\|_2$ is the spectral condition number of X. The eigenvectors are also computed accurately in the appropriate sense.

We believe that this is the first algorithm to compute accurate eigenvalues of symmetric (indefinite) matrices that respects and preserves the symmetry of the problem and uses only orthogonal transformations.

Keywords eigenvalues · eigenvectors · high relative accuracy · Jacobi algorithm

Mathematics Subject Classification (2000) 65F15 · 65G50 · 15A23

1 Introduction

When conventional algorithms, like *QR* or divide-and-conquer, are used to compute the eigenvalues and eigenvectors of ill-conditioned *real symmetric matrices* in floating point arithmetic, only the largest in magnitude eigenvalues are computed with guaranteed relative accuracy. The tiny eigenvalues may be computed with no relative accuracy at all—and even

F. M. Dopico and J. M. Molera were partially supported by the Ministerio de Educación y Ciencia of Spain through grant MTM-2006-06671 and by the PRICIT Program of the Comunidad de Madrid through SIMUMAT Project (Ref. S-0505/ESP/0158). P. Koev was partially supported by National Science Foundation grants DMS–0314286 and DMS–0608306.

Froilán M. Dopico
Instituto de Ciencias Matemáticas CSIC-UAM-UC3M-UCM and Departamento de Matemáticas, Universidad Carlos III de Madrid, Avda. Universidad 30, 28911 Leganés, Spain. E-mail: dopico@math.uc3m.es

Plamen Koev
Department of Mathematics, San Jose State University, One Washington Square, San Jose, CA 95192, United States. E-mail: koev@math.sjsu.edu

Juan M. Molera
Departamento de Matemáticas, Universidad Carlos III de Madrid, Avda. Universidad 30, 28911 Leganés, Spain. E-mail: molera@math.uc3m.es

with the wrong sign. The only eigenvectors that are computed accurately are the ones corresponding to eigenvalues whose absolute separation from the rest of the spectrum is large. See [2, section 4.7] for a survey on error bounds for the symmetric eigenproblem.

In contrast, in the last twenty years an intensive research effort has been made to derive algorithms for computing eigenvalues and eigenvectors of $n \times n$ symmetric matrices *to high relative accuracy*, at $O(n^3)$ cost, i.e., roughly the same cost as that of conventional algorithms for dense symmetric matrices [3,12,14,16,31,32,35,38,41,42]. The closely related problem of computing the Singular Value Decomposition (SVD) with high relative accuracy has received even more attention [5–8,10,11,18,20,21,23,45].

By *high relative accuracy* we mean that the eigenvalues λ_i, the eigenvectors v_i, and their computed counterparts $\hat{\lambda}_i$ and $\hat{v}_i$, respectively, satisfy

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

where ε is the machine precision, and $\theta(v_i, \hat{v}_i)$ is the acute angle between v_i and $\hat{v}_i$. These conditions guarantee that *all* eigenvalues, including the tiniest ones, are computed with correct sign and leading digits. The eigenvectors are computed accurately as long as the *relative* separations between the eigenvalues are large, regardless of how small the eigenvalues themselves may be. For multiple or extremely close eigenvalues, the eigenvectors become extremely ill conditioned in which case we develop error bounds for the corresponding invariant subspaces.

Many classes of structured symmetric matrices whose eigendecompositions and SVDs can be computed with high relative accuracy have been identified [3,5–12,14,38,45]. Introduced by Demmel et al. [7], the key unifying idea in these high accuracy computations is to compute first an accurate *rank revealing decomposition* (RRD), i.e., a decomposition $A = XDX^T$, where X is well conditioned and D is diagonal, and then to recover the eigenvalues and eigenvectors from the factors of the RRD.

At present, accurate computations of RRDs are possible for the following classes of *symmetric* matrices: scaled diagonally dominant matrices [3], diagonally scaled well conditioned positive definite matrices [12], certain diagonally scaled well conditioned indefinite matrices [40], weakly diagonally dominant M-matrices [10], Cauchy matrices, diagonally scaled Cauchy matrices, Vandermonde matrices, totally nonnegative matrices [14], total signed compound matrices, diagonally scaled totally unimodular matrices [38], and properly parameterized diagonally dominant matrices [45].

The fundamental property that makes an RRD very useful in high relative accuracy computations is that its factors accurately determine the eigenvalues and eigenvectors of the original matrix. Namely, small componentwise relative perturbations in D and small normwise relative perturbations in X produce small relative perturbations in the eigenvalues of A, and small perturbations in the eigenvectors with respect to the relative eigenvalue gap [7,14].

Several algorithms have been proposed in the past to compute eigendecompositions of symmetric RRDs to high relative accuracy. These algorithms are very satisfactory in the positive definite case and are based on the one-sided Jacobi algorithm [13, Section 5.4.3] with a stringent stopping criterion [12,18,35].

Two algorithms are proposed for the indefinite case. While both algorithms work well in practice, they both have shortcomings:

- The algorithm proposed by Veselić [44], and carefully analyzed and developed by Slapničar [41,42] uses hyperbolic transformations, an unfortunate situation, since symmetric matrices are diagonalizable by an orthogonal similarity. Furthermore, this hyperbolic procedure does not guarantee small error bounds[1];

- In contrast, the algorithm of Dopico, Molera, and Moro [16] does guarantee the error bounds (1), but does not respect the symmetry of the problem.

Our main result in this paper is a new algorithm which, given an RRD $A = XDX^T$ of a symmetric matrix A (definite or indefinite), computes its eigenvalues and eigenvectors to high relative accuracy by *using only orthogonal transformations and respecting the symmetry of the problem.* When A is nonsingular this algorithm is simply the standard Jacobi algorithm applied implicitly on X using the well known cyclic-by-row strategy [13, Section 5.3.5] to create the "implicit" zeros in A. The algorithm stops when

$$|a_{ij}| \leq \text{tol}\sqrt{|a_{ii}a_{jj}|}, \tag{2}$$

for all $i < j$, where tol is a given tolerance, typically $O(\varepsilon)$. Once the stopping criterion has been satisfied, the eigenvalues of A are computed as the diagonal entries of $X_f D X_f^T$, where X_f is the last iterate. The eigenvectors are accumulated from the Jacobi rotations in each step.

In Section 5 we prove that the relative error in each eigenvalue is bounded by $O(\kappa(X)\varepsilon)$, where $\kappa(X) \equiv \|X\|_2\|X^{-1}\|_2$ is the condition number of X, and $\|\cdot\|_2$ is the spectral norm. Note that since we are using only orthogonal transformations in the Jacobi iteration, the condition number of X does not change. Therefore, when X is well conditioned, i.e., when $\kappa(X) \ll \frac{1}{\varepsilon}$, the eigenvalues are computed to high relative accuracy. Roughly, each eigenvalue is computed with $\log_{10} 1/(\varepsilon\kappa(X))$ correct leading significant decimal digits. We also prove that the error in each computed eigenvector is bounded by $O(\kappa(X)\varepsilon)$ divided by the corresponding relative eigenvalue gap.

To establish our relative error bounds we prove that the computed eigenvalues and eigenvectors are the exact eigenvalues and eigenvectors of a *small multiplicative perturbation of* XDX^T, i.e.,

$$(I+E)XDX^T(I+E)^T, \tag{3}$$

with $\|E\|_2 = O(\varepsilon\,\kappa(X))$. This backward error result is in stark contrast with the *unstructured additive* backward error bounds for conventional symmetric eigensolvers [2, section 4.7]. This is the key fact which, combined with the multiplicative perturbation theory bounds in [22,33,34], allows us to prove that the implicit Jacobi algorithm delivers high relative accuracy when X is well conditioned.

In exact arithmetic, the implicit Jacobi algorithm is mathematically equivalent to applying the standard cyclic-by-row Jacobi algorithm to XDX^T, and therefore the convergence properties of the implicit method are the same as those of standard Jacobi, to be found for instance in [13,25,37]. Thus we do not address its convergence properties in this paper.

The paper is organized as follows. We introduce the main result—the implicit Jacobi algorithm for nonsingular RRDs—in Section 2, as well as some of its key properties. It sets the stage for the rest of the paper, where detailed proofs and tests are developed. In Section 3 we summarize multiplicative perturbation bounds for eigenvalues and eigenvectors. Section 4 is concerned with the accuracy of the last step of the implicit Jacobi algorithm. In Section 5 a complete multiplicative backward error analysis for the implicit Jacobi algorithm is

[1] See [41, Theorem 4] and do notice that the error bound for the eigenvalues depends on the inverse of the minimum singular values of all the *column scaled* matrix iterates generated by the hyperbolic one-sided Jacobi algorithm. As far as we know, there is no proof that these quantities are bounded, but they have never been observed to be large in practice.

developed. In Section 6 we show that our algorithm extends trivially to singular RRDs. Since the factors X and D of an RRD may be results of previous computations (and thus carry uncertainties), we discuss how these uncertainties affect the final output in Section 7. In Section 8 we present a simple but very effective preconditioning technique to speed up the implicit Jacobi algorithm. We present numerical tests in Section 9 and draw conclusions in Section 10.

Notation: In this paper we consider only real matrices and denote the set of $m \times n$ real matrices by $\mathbb{R}^{m\times n}$. The entries of a matrix A are denoted by a_{ij} and $|A|$ is the matrix with entries $|a_{ij}|$. We use MATLAB [36] 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.

2 The implicit Jacobi algorithm

In this section we present our main result—the implicit Jacobi algorithm. We assume that an RRD $A = XDX^T$ of a symmetric matrix A is given, where $X, D \in \mathbb{R}^{n\times n}$ are nonsingular and $D = \operatorname{diag}(d_1,\dots,d_n)$. The case when X is rectangular or D is singular is considered in Section 6. Note that when A is nonsingular its eigenvalues are different from zero, therefore the Jacobi algorithm stops in a final iterate that is an almost diagonal matrix with nonzero diagonal entries.

We adopt the standard notation for Jacobi rotations

$$R(i,j,c,s)=\begin{array}{cc} & \begin{array}{ccccccc} & & i & & j & & \end{array} \\ \begin{array}{c} \\ \\ i \\ \\ j \\ \\ \\ \end{array} & \left[\begin{array}{ccccccc} 1 &&&&&& \\ & \ddots &&&&& \\ && c && -s && \\ &&& \ddots &&& \\ && s && c && \\ &&&&& \ddots & \\ &&&&&& 1 \end{array}\right] \end{array},$$

where the computation of the cosines, c, and sines, s, is performed in the traditional way – see the classical texts [13, Section 5.3.5], [25, Section 8.4.2], and [37, Chapter 9] for details.

The key idea of our algorithm below is to apply the Jacobi rotations implicitly and keep the matrix in factored form, i.e., to implement each Jacobi step $XDX^T \to R^T(XDX^T)R$ by updating $X \to R^TX$. Since the multiplicative backward errors introduced by this update are unaffected by right diagonal scaling on X, we refactor XDX^T as GJG^T, where $G = X\operatorname{diag}(\sqrt{|d_1|},\dots,\sqrt{|d_n|})$ and $J = \operatorname{diag}(\operatorname{sign}(d_1),\dots,\operatorname{sign}(d_n))$, and update G instead. We use this second updating procedure because it is more convenient in the preconditioned version in Algorithm 3.

The entries of $A = GJG^T$ needed for the computation of the Jacobi rotation in Algorithm 1 and for the computation of the eigenvalues in the last step are computed through the usual formula

$$a_{ij} = \sum_{k=1}^{n} g_{ik}g_{jk}\operatorname{sign}(d_k). \tag{4}$$

Algorithm 1, and the rest of algorithms in this paper, guarantees high relative accuracy in the computed eigenvalues and eigenvectors if X is well conditioned. As explained in the Introduction, this means that $\kappa(X) \ll \frac{1}{\varepsilon}$. We will see that the smaller the condition number of X, the larger the accuracy.

Algorithm 1 (**Implicit cyclic-by-row Jacobi on $\mathbf{XDX^T}$**) Given a nonsingular well conditioned matrix $X \in \mathbb{R}^{n\times n}$ and a diagonal nonsingular matrix $D = \operatorname{diag}(d_1,\ldots,d_n) \in \mathbb{R}^{n\times n}$, this algorithm computes the eigenvalues $\lambda_1,\ldots,\lambda_n$ of $A = XDX^T$ and an orthogonal matrix $U \in \mathbb{R}^{n\times n}$ of eigenvectors to high relative accuracy.

```
$\widehat{\kappa}(X)$ is the computed estimation of $\kappa(X)$
$U = I_n$
$G = X \operatorname{diag}(\sqrt{|d_1|},\ldots,\sqrt{|d_n|})$
$J = \operatorname{diag}(\operatorname{sign}(d_1),\ldots,\operatorname{sign}(d_n))$
repeat
    for $i = 1 : n-1$
        for $j = i+1 : n$
            compute $a_{ii}, a_{ij}, a_{jj}$ of $A = GJG^T$ as in (4)
            compute $T = \begin{bmatrix} c & -s \\ s & c \end{bmatrix}$, $c^2+s^2=1$, such that $T^T \begin{bmatrix} a_{ii} & a_{ij} \\ a_{ij} & a_{jj} \end{bmatrix} T = \begin{bmatrix} \mu_1 & 0 \\ 0 & \mu_2 \end{bmatrix}$
            $G = R(i,j,c,s)^T G$
            $U = U R(i,j,c,s)$
        endfor
    endfor
until convergence $\left( \frac{|a_{ij}|}{\sqrt{|a_{ii}a_{jj}|}} \leq \varepsilon \max\{n, \widehat{\kappa}(X)\}$ for all $i<j$ and $\frac{\sum_{k=1}^n g_{ik}^2}{|a_{ii}|} \leq 2\widehat{\kappa}(X)$ for all $i \right)$
compute $\lambda_i = a_{ii}$ for $i = 1,2,\ldots,n$.
```

Apart from the implicit nature of Algorithm 1, it differs from the usual Jacobi algorithm in the last two lines—the stopping criterion and the final computation of the eigenvalues. Both lines paramount in guaranteeing high relative accuracy of the computed eigenvalues and the associated error analysis is one of the main contributions in this work. They deserve a brief explanation.

Let us start with the last line of the code, the computation of the eigenvalues as the diagonal entries of the last iterate $A = GJG^T$ from the factors G and J using (4). To get the eigenvalues with high relative accuracy it is necessary to guarantee that no severe cancelation is produced in this process. To this purpose, first we will prove in Theorem 5 in Section 4 that, in exact arithmetic, if the implicit Jacobi algorithm stops according with the usual stopping criterion (2), then the conditions

$$\frac{\sum_{k=1}^n g_{ik}^2}{|a_{ii}|} \leq 2\kappa(X), \quad i = 1,\ldots,n, \tag{5}$$

are automatically satisfied for the last iterate. This fact is what induces the use of (5) as the second part of the stopping criterion in the line before the last in Algorithm 1. This second part of the criterion, together with standard error analysis [29, Section 3.1], leads to the following satisfactory relative error bounds in the a_{ii} computed in the last line:

$$\left| \frac{fl(a_{ii}) - a_{ii}}{a_{ii}} \right| \leq \frac{n\varepsilon}{1-n\varepsilon} \cdot \frac{\sum_{k=1}^n g_{ik}^2}{|a_{ii}|} \leq \frac{2n\varepsilon}{1-n\varepsilon} \widehat{\kappa}(X), \quad i = 1,\ldots,n. \tag{6}$$

These relative errors are small whenever $\kappa(X)$ is small. Note that in exact arithmetic the conditions (5) are satisfied without the need of extra Jacobi steps with respect to (2). We have always observed the same in thousands of numerical tests, but, in finite precision, we need to impose (5) explicitly to guarantee the error bounds.

The stopping criterion (2) with $\text{tol} = \varepsilon \max\{n, \widehat{\kappa}(X)\}$ in Algorithm 1 includes $\varepsilon\widehat{\kappa}(X)$ because $\frac{|a_{ij}|}{\sqrt{|a_{ii}a_{jj}|}}$ involves the computed entries a_{ii}, a_{jj} and a_{ij}, and (6) imply relative errors of order $\varepsilon\widehat{\kappa}(X)$ in the computed diagonal entries. A complete explanation of the stopping criterion in Algorithm 1 is presented in Section 5.1

The crucial part of the error analysis of Algorithm 1 corresponds to the stopping criterion because standard error analysis guarantees that the application of the Jacobi rotations on G is safe. The reason is that G is well conditioned after column scaling since X is well conditioned, and therefore only small backward multiplicative errors are introduced by the rotations [13, p. 251] [29, Lemma 19.9]. In fact, we will see in Lemma 3 in Section 5.1 that the errors introduced by the stopping criterion can also be expressed as small backward multiplicative errors. This is combined with the errors coming from the rotations in Theorem 6 in Section 5.2 to prove that Algorithm 1 computes the eigenvalues and eigenvectors of XDX^T with small multiplicative backward errors (3) with $\|E\|_2 = O(\varepsilon\kappa(X))$. We will recall in Section 3 multiplicative perturbation results for eigenvalues and invariant subspaces (eigenvectors) that together with Theorem 6 show that Algorithm 1 computes the eigenvalues and eigenvectors of XDX^T with errors

$$|\hat{\lambda}_i - \lambda_i| \leq O(\varepsilon\kappa(X))|\lambda_i| \quad \text{and} \quad \theta(v_i, \hat{v}_i) \leq \frac{O(\varepsilon\,\kappa(X))}{\min\limits_{j\neq i}\left|\frac{\lambda_i-\lambda_j}{\lambda_i}\right|} \quad \text{for} \quad i = 1, \ldots, n. \tag{7}$$

In the case of extremely close or equal eigenvalues, the bound for $\theta(v_i, \hat{v}_i)$ explodes, then one can get bounds for the invariant subspaces using Theorem 2.

Let us consider the computational cost of Algorithm 1. Assume, without loss of generality, that the p positive entries on the diagonal of D come first, then the entries a_{ij} are $a_{ij} = \sum_{k=1}^{p} g_{ik}g_{jk} - \sum_{k=p+1}^{n} g_{ik}g_{jk}$. So the cost of computing a Jacobi rotation is $6n$ flops, the cost of multiplying G by a Jacobi rotation is $6n$ flops, and the cost of multiplying U by a Jacobi rotation is $6n$ flops. So each Jacobi step costs $12n$ flops if the eigenvectors are not desired and $18n$ if they are. If N_R is the total number of Jacobi rotations performed in Algorithm 1 the total cost is

$$\begin{array}{lll} 12\,nN_R & \text{flops} & \text{to compute only eigenvalues} \\ 18\,nN_R & \text{flops} & \text{to compute eigenvalues and eigenvectors.} \end{array}$$

These costs can also be expressed in terms of the number of Jacobi sweeps, denoted by N_{sw}. Since one sweep involves $n(n-1)/2$ consecutive rotations, the cost is $6\,n^3\,N_{sw}$ flops to compute eigenvalues and $9\,n^3\,N_{sw}$ if the eigenvectors are also desired. It is accepted that N_{sw} is proportional to $\log(n)$ for the classical Jacobi algorithm [25, Section 8.4]. For information on the number of sweeps performed by Algorithm 1 we refer to the numerical tests presented in Section 9. Several ideas to decrease the computational cost of Algorithm 1 are discussed in Section 8.

In practice, the rotation $R(i, j, c, s)$ is applied on G and/or U only if

$$|a_{ij}| > \varepsilon \max\{n, \widehat{\kappa}(X)\}\sqrt{|a_{ii}a_{jj}|},$$

or

$$\sum_{k=1}^{n} g_{ik}^2 > 2\widehat{\kappa}(X)\,|a_{ii}|, \quad \text{or} \quad \sum_{k=1}^{n} g_{jk}^2 > 2\widehat{\kappa}(X)\,|a_{jj}|.$$

Once a_{ij}, a_{ii}, and a_{jj} are computed, the cost of checking the first condition is 3 flops, negligible with respect to the cost of a rotation. The second condition involves the unknown quantity $\sum_{k=1}^{n} g_{ik}^2$ that can be computed with negligible cost as follows: assume again that the p positive entries on the diagonal of D come first, then we can compute $a_{ii} = \sum_{k=1}^{p} g_{ik}^2 - \sum_{k=p+1}^{n} g_{ik}^2$ and $\sum_{k=1}^{n} g_{ik}^2 = \sum_{k=1}^{p} g_{ik}^2 + \sum_{k=p+1}^{n} g_{ik}^2$. As a consequence, to compute $\sum_{k=1}^{n} g_{ik}^2$ only costs one additional flop if $\sum_{k=1}^{p} g_{ik}^2$ and $\sum_{k=p+1}^{n} g_{ik}^2$ are stored. A similar remark holds for $\sum_{k=1}^{n} g_{jk}^2$. So, the total cost of checking the second condition of the stopping criterion is four flops.

Finally, we mention that if the matrix D is extremely ill-conditioned, then underflow may appear in the computation of the Jacobi rotations. This can cause loss of accuracy and failure of convergence. In this case, the rotations should be carefully implemented in the spirit of the procedure presented in [17] for the SVD computation.

3 Basic results on multiplicative perturbation theory

In this section we recall two bounds for the relative perturbations of eigenvalues and eigenvectors of symmetric matrices under multiplicative perturbations [22,34].

Theorem 1 [22, Theorem 2.1] *Let $A = A^T \in \mathbb{R}^{n\times n}$ and $\widetilde{A} = (I+E)A(I+E)^T \in \mathbb{R}^{n\times n}$, where $I+E$ is nonsingular. Let $\lambda_1 \geq \cdots \geq \lambda_n$ and $\widetilde{\lambda}_1 \geq \cdots \geq \widetilde{\lambda}_n$ be, respectively, the eigenvalues of A and $\widetilde{A}$. Then*

$$|\widetilde{\lambda}_i - \lambda_i| \leq (2\,\|E\|_2 + \|E\|_2^2)\,|\lambda_i|, \quad \textit{for } i = 1,\ldots,n.$$

Lemma 1 is a corollary of Theorem 1.

Lemma 1 *Let $X \in \mathbb{R}^{n\times r}$ be a matrix of full column rank, and $D = \mathrm{diag}(d_1,\ldots,d_r)$ and $\widetilde{D} = \mathrm{diag}(\widetilde{d}_1,\ldots,\widetilde{d}_r) \in \mathbb{R}^{r\times r}$ be nonsingular diagonal matrices such that*

$$|\widetilde{d}_i - d_i| \leq \beta\,|d_i|, \qquad i = 1,\ldots r,$$

where $0 \leq \beta < 1$. Let $\lambda_1 \geq \cdots \geq \lambda_n$ and $\widetilde{\lambda}_1 \geq \cdots \geq \widetilde{\lambda}_n$ be, respectively, the eigenvalues of XDX^T and $X\widetilde{D}X^T$. Define $\alpha = 1 - \sqrt{1-\beta}$, and assume that $\alpha\kappa(X) < 1$. Then

$$|\widetilde{\lambda}_i - \lambda_i| \leq |\lambda_i|\,(2 + \alpha\kappa(X))\,\alpha\kappa(X), \qquad i = 1,\ldots.,n.$$

Proof Let $\widetilde{d}_i = d_i(1+\mu_i)$, where $|\mu_i| \leq \beta$, for $i = 1,\ldots r$. We define δ_i from $1+\delta_i \equiv \sqrt{1+\mu_i}$. Then $|\delta_i| \leq 1 - \sqrt{1-\beta} = \alpha$. We also define $\Delta = \mathrm{diag}(\delta_1,\ldots,\delta_r)$, obtaining

$$\widetilde{D} = (I+\Delta)D(I+\Delta)^T, \quad \|\Delta\|_2 \leq \alpha.$$

If $X^\dagger$ is the pseudoinverse of X then $X^\dagger X = I$, therefore

$$X\widetilde{D}X^T = X(I+\Delta)D(I+\Delta)^T X^T = (I + X\Delta X^\dagger)XDX^T(I + X\Delta X^\dagger)^T, \tag{8}$$

and $\|X\Delta X^\dagger\|_2 \leq \alpha\kappa(X) < 1$. Therefore $I + X\Delta X^\dagger$ is nonsingular, and Theorem 1 can be applied to (8) to obtain the result. □

For the eigenvector perturbations we use the results of Li [34]. The presence of multiple or extremely close eigenvalues is permitted by bounding the canonical angles [43] between invariant subspaces. We establish some additional notation to state the perturbation bound. Let A and $\widetilde{A}$ be two real $n \times n$ symmetric matrices with eigendecompositions

$$A = [U_1\ U_2] \begin{bmatrix} \Lambda_1 & \\ & \Lambda_2 \end{bmatrix} \begin{bmatrix} U_1^T \\ U_2^T \end{bmatrix} \quad \text{and} \quad \widetilde{A} = [\widetilde{U}_1\ \widetilde{U}_2] \begin{bmatrix} \widetilde{\Lambda}_1 & \\ & \widetilde{\Lambda}_2 \end{bmatrix} \begin{bmatrix} \widetilde{U}_1^T \\ \widetilde{U}_2^T \end{bmatrix}, \tag{9}$$

where $U_1, \widetilde{U}_1 \in \mathbb{R}^{n\times k}$, $[U_1\ U_2]$ and $[\widetilde{U}_1\ \widetilde{U}_2]$ are $n \times n$ orthogonal matrices, and $\Lambda_1, \Lambda_2, \widetilde{\Lambda}_1$, and $\widetilde{\Lambda}_2$ are diagonal matrices. We denote by $\Theta(U_1, \widetilde{U}_1)$ the canonical angles between $\mathrm{Span}(U_1)$ and $\mathrm{Span}(\widetilde{U}_1)$, and by $\lambda(\widetilde{\Lambda}_1)$ and $\lambda(\widetilde{\Lambda}_2)$ the spectra of $\widetilde{\Lambda}_1$ and $\widetilde{\Lambda}_2$, respectively. We assume in (9) that if the eigenvalues of A and $\widetilde{A}$ are decreasingly ordered, i.e., $\lambda_1 \geq \cdots \geq \lambda_n$ and $\widetilde{\lambda}_1 \geq \cdots \geq \widetilde{\lambda}_n$, then $\widetilde{\Lambda}_1 = \{\widetilde{\lambda}_{i_1}, \ldots, \widetilde{\lambda}_{i_k}\}$ if and only if $\Lambda_1 = \{\lambda_{i_1}, \ldots, \lambda_{i_k}\}$.

Theorem 2 [34, Theorem 2.2, Remark 2.1] *Let $A = A^T \in \mathbb{R}^{n\times n}$ and*

$$\widetilde{A} = (I+E)A(I+E)^T \in \mathbb{R}^{n\times n},$$

where $\|E\|_2 < 1$, have eigendecompositions (9). *Let us assume that $\lambda(\widetilde{\Lambda}_1) \cap \lambda(\widetilde{\Lambda}_2) = \emptyset$ and define*

$$\mathrm{relgap}(\widetilde{\Lambda}_1) = \min\left(\min_{\mu\in\lambda(\widetilde{\Lambda}_1), \nu\in\lambda(\widetilde{\Lambda}_2)} \frac{|\mu-\nu|}{|\mu|}, 1 \right).$$

Then

$$\frac{1}{2}\|\sin 2\Theta(U_1, \widetilde{U}_1)\|_F \leq \frac{2\sqrt{k}}{\mathrm{relgap}(\widetilde{\Lambda}_1)} \cdot \frac{1+\|E\|_2}{1-\|E\|_2} (2\|E\|_2 + \|E\|_2^2),$$

where $\|\cdot\|_F$ denotes the Frobenius matrix norm.

The case for single eigenvectors corresponds to $k = 1$.

4 Diagonal and scaled diagonally dominant RRDs

In this section we focus on RRDs $A = XDX^T$ such that X and D are nonsingular square matrices. We will consider the singular and rectangular cases in Section 6.

In the last step of Algorithm 1 the eigenvalues are computed as the diagonal entries of a matrix satisfying the stopping criterion (2). We will call the matrices satisfying (2) *scaled diagonally dominant* matrices[2]. According to (6) the diagonal entries of such matrices can be safely and accurately computed in floating point arithmetic from the factors of the RRD $A = XDX^T = GJG^T$ through the formula $a_{ii} = \sum_{k=1}^n g_{ik}^2 \mathrm{sign}(d_k)$, $i = 1, \ldots, n$, if the ratios

$$\frac{\sum_{k=1}^n g_{ik}^2}{\left|\sum_{k=1}^n g_{ik}^2 \,\mathrm{sign}(d_k)\right|} = \frac{\sum_{k=1}^n x_{ik}^2 |d_k|}{\left|\sum_{k=1}^n x_{ik}^2 d_k\right|} = \frac{\sum_{k=1}^n x_{ik}^2 |d_k|}{|a_{ii}|}, \quad i = 1, \ldots, n, \tag{10}$$

[2] Note that a matrix A with nonzero diagonal entries and satisfying the stopping criterion (2) can be expressed as $A = D_A C D_A$, where $D_A = \mathrm{diag}(\sqrt{|a_{11}|}, \ldots, \sqrt{|a_{nn}|})$ and $|c_{ij}| \leq \mathrm{tol}$ if $i \neq j$. Therefore, according to the definition in [3, pp. 764-765], A is tol-scaled diagonally dominant with respect to the (non-consistent) max-norm, i.e., $\|B\|_M \equiv \max_{ij} |b_{ij}|$ for any matrix B. This is the reason why we adopt the name *scaled diagonally dominant* for matrices satisfying (2). For brevity, we omit the norm and the parameter tol.

are much smaller than $1/(n\varepsilon)$. The purpose of this section is to prove in exact arithmetic that the ratios in (10) are essentially bounded by $\kappa(X)$ when the matrix A fulfils (2).

We first consider diagonal RRDs in Theorem 3, then the final result for scaled diagonally dominant RRDs is proved in Theorem 5.

Theorem 3 *Let $X \in \mathbb{R}^{n\times n}$ be nonsingular and $D = \operatorname{diag}(d_1,\dots,d_n)$ be diagonal and nonsingular. If XDX^T is diagonal then*

$$\frac{\displaystyle\sum_{k=1}^{n} x_{ik}^2 |d_k|}{\left|\displaystyle\sum_{k=1}^{n} x_{ik}^2 d_k\right|} \leq \kappa(X), \quad i = 1,2,\dots,n. \tag{11}$$

Proof We denote $\Lambda = XDX^T$, where $\Lambda = \operatorname{diag}(\lambda_1,\dots,\lambda_n)$. Note that $\lambda_i = \sum_{k=1}^n x_{ik}^2 d_k$, $i = 1,\dots,n$, are the eigenvalues of XDX^T and that, for each i, the left and right eigenvectors of λ_i are both equal to the ith column of I_n. We denote this ith column by e_i. We assume without loss of generality that $\lambda_1 \geq \cdots \geq \lambda_n$.

First, we prove the result when λ_i is a simple eigenvalue. Let $\widetilde{d_j} \equiv d_j\,(1+\delta\,\operatorname{sign}(d_j)) = d_j + \delta|d_j|$, for $j = 1,\dots,n$, where δ is a small real parameter, and let $\widetilde{D} \equiv \operatorname{diag}(\widetilde{d_1},\dots,\widetilde{d_n})$. Let $\widetilde{\lambda}_1 \geq \cdots \geq \widetilde{\lambda}_n$ be the eigenvalues of $X\widetilde{D}X^T = \Lambda + \delta X|D|X^T$. The classical first order perturbation expansion for simple eigenvalues [13, Theorem 4.4] implies[3] that as $\delta \to 0$

$$\widetilde{\lambda}_i = \lambda_i + \delta\, e_i^T X|D|X^T e_i + O(\delta^2) = \lambda_i + \delta \sum_{k=1}^{n} x_{ik}^2 |d_k| + O(\delta^2).$$

Then

$$\lim_{\delta\to 0} \frac{1}{\delta}\,\frac{\widetilde{\lambda}_i - \lambda_i}{\lambda_i} = \frac{\sum_{k=1}^n x_{ik}^2 |d_k|}{\lambda_i}. \tag{12}$$

On the other side, Lemma 1 can be applied to XDX^T and $X\widetilde{D}X^T$ with $\beta = |\delta|$ and $\alpha = 1-\sqrt{1-\beta} = |\delta|/2 + O(\delta^2)$ to get

$$\frac{|\widetilde{\lambda}_i - \lambda_i|}{|\lambda_i|} \leq |\delta|\kappa(X) + O(\delta^2)$$

and

$$\lim_{\delta\to 0} \frac{1}{|\delta|}\,\frac{|\widetilde{\lambda}_i - \lambda_i|}{|\lambda_i|} \leq \kappa(X).$$

This is combined with (12) to prove the claim.

Next, let λ_i be a multiple eigenvalue, e.g., $\lambda_{l-1} > \lambda_l = \cdots = \lambda_i = \cdots = \lambda_p > \lambda_{p+1}$. Pick $\delta > 0$ and define $D' \equiv \operatorname{diag}(d'_1,\dots,d'_n)$, where $d'_k = 1+\delta$ if $l \leq k \leq (i-1)$ or $(i+1) \leq k \leq p$, and $d'_k = 1$ otherwise. Note that $d'_i = 1$, therefore $(D'X)_{ik} = x_{ik}$ for all k, and λ_i is a *simple* eigenvalue of the diagonal matrix

$$D'\Lambda D' = (D'X)D(D'X)^T.$$

[3] According to [13, Theorem 4.4], one gets $\widetilde{\lambda}_i = \lambda_i + \delta\, e_i^T X|D|X^T e_i + O(\|\delta X|D|X^T\|_2^2)$. The exact meaning of $O(\|\delta X|D|X^T\|_2^2)$ is that there exists a constant K depending only on XDX^T and not on the perturbation such that $|O(\|\delta X|D|X^T\|_2^2)| \leq K\|\delta X|D|X^T\|_2^2 \leq K\|X\|_2^4\|D\|_2^2\delta^2$. Note that this is $O(\delta^2)$ with the constant $K\|X\|_2^4\|D\|_2^2$, that depends only on the unperturbed matrix XDX^T and not on the perturbation. Since the value of the constant is not relevant in our argument, we simply use $O(\delta^2)$.

Thus, we can apply the result for simple eigenvalues to the ith entry of the diagonal matrix $(D'X)D(D'X)^T$ to get

$$\frac{\sum_{k=1}^{n} x_{ik}^2 |d_k|}{|\lambda_i|} \leq \kappa(D'X),$$

where the left hand side does not depend on δ. By taking the limit $\delta \to 0$ the result follows. □

Next, we prove an auxiliary result—that setting the off-diagonal entries of a scaled diagonally dominant matrix to zero is a small multiplicative perturbation of the matrix.

Theorem 4 *Let $A = A^T \in \mathbb{R}^{n\times n}$ be such that $a_{ii} \neq 0$ for all i, and*

$$\frac{|a_{ij}|}{\sqrt{|a_{ii}a_{jj}|}} \leq \delta \quad \textit{for all } \; i \neq j, \tag{13}$$

where $\delta \leq \frac{1}{5n}$. Then the following hold.

1. $\operatorname{diag}(a_{11},\dots,a_{nn}) = (I+F)A(I+F)^T$ *with* $\|F\|_F \leq \dfrac{n\delta}{1-2n\delta}$.
2. *Let $D_A \equiv \operatorname{diag}(\sqrt{|a_{11}|},\dots,\sqrt{|a_{nn}|})$. If $|a_{11}| \geq \cdots \geq |a_{nn}|$, then*

$$A = (I+E)\operatorname{diag}(a_{11},\dots,a_{nn})\,(I+E)^T,$$

where E is lower triangular, $\left\|D_A^{-1}ED_A\right\|_F \leq \dfrac{n\delta}{1-n\delta}$, and $\left\|D_A^{-1}ED_A\right\|_\infty \leq \dfrac{5}{4}\dfrac{n\delta}{1-n\delta}$. Here $\|\cdot\|_\infty$ denotes the ∞-matrix norm [29, p. 108].

Proof The result in the first claim is invariant under permutations PAP^T of A, thus we assume that $|a_{11}| \geq \cdots \geq |a_{nn}|$. Define $C \equiv D_A^{-1}AD_A^{-1}$ and note that C is symmetric, $c_{ii} = \operatorname{sign}(a_{ii})$ for all i, and $|c_{ij}| \leq \delta$ for all $i \neq j$. Let $J \equiv \operatorname{diag}(c_{11},\dots,c_{nn})$ and write

$$C = J+G, \quad \text{where } \|G\|_F \leq n\delta \leq \frac{1}{5} \quad \text{and} \quad \|G\|_\infty \leq n\delta \leq \frac{1}{5}. \tag{14}$$

Next, we prove that C has a unique LU factorization with the factor L unit lower triangular by showing that all its leading principal submatrices are nonsingular [29, Theorem 9.1]. Let B_k denote the kth leading principal submatrix of any matrix B. Then, since J_k is nonsingular, $C_k = J_k + G_k$ is nonsingular if and only if $J_kC_k = I + J_kG_k$ is nonsingular. The matrix $I+J_kG_k$ is nonsingular because $\|J_kG_k\|_F = \|G_k\|_F \leq \|G\|_F < 1$. Since C is symmetric, it has a unique LDL^T factorization:

$$C = \bar{L}\bar{D}\bar{L}^T, \tag{15}$$

again with the factor $\bar{L}$ unit lower triangular. Equation (14) allows us to consider C as a perturbation of J, where the unique LDL^T factors of J are simply $L = I$ and $D = J$. Then Theorem 6.2 in [15] can be used to get[4]

$$|\bar{L}-I| \leq \left(|G|(I-|G|)^{-1}\right)_L \quad \text{and} \tag{16}$$

$$|\bar{D}-J| \leq \left(|G|(I-|G|)^{-1}\right)_D, \tag{17}$$

[4] Theorem 6.2 in [15] holds for block LDL^T factorizations. In our case all blocks are 1×1.

where for any matrix B, $(B)_L$ denotes its strictly lower triangular part and $(B)_D$ its diagonal part. Note that the matrix $|G|(I-|G|)^{-1} = \sum_{k=1}^{\infty} |G|^k$ is symmetric, because $|G|$ is. Therefore the bound (16) implies

$$\|\bar{L}-I\|_F \leq \| \left(|G|(I-|G|)^{-1}\right)_L \|_F \leq \frac{1}{\sqrt{2}} \||G|(I-|G|)^{-1}\|_F \leq \frac{1}{\sqrt{2}} \cdot \frac{\|G\|_F}{1-\|G\|_F}, \qquad (18)$$

for the Frobenius norm, and

$$\|\bar{L}-I\|_\infty \leq \||G|(I-|G|)^{-1}\|_\infty \leq \frac{\|G\|_\infty}{1-\|G\|_\infty}, \qquad (19)$$

for the ∞-norm.

The next step is to use the bound (17) to prove that

$$\bar{D} = (I+D')J(I+D')^T, \quad \text{where } \|D'\|_F \leq \frac{\|G\|_F^2}{1-\|G\|_F} \quad \text{and} \quad \|D'\|_\infty \leq \frac{\|G\|_\infty^2}{1-\|G\|_\infty}, \qquad (20)$$

and $D' = \operatorname{diag}(d_1', \ldots, d_n')$ is diagonal. For this purpose, we write $\bar{D} = J + \bar{D} - J = J(I+W)$, where $W = \operatorname{diag}(w_1, \ldots, w_n) \equiv J(\bar{D} - J)$. Thus, from (17),

$$|W| \leq \sum_{k=1}^{\infty} \left(|G|^k\right)_D = \sum_{k=2}^{\infty} \left(|G|^k\right)_D,$$

because $(|G|)_D = 0$ by (14). Therefore every diagonal entry of W satisfies,

$$|w_i| \leq \|W\|_F \leq \sum_{k=2}^{\infty} \left\| \left(|G|^k\right)_D \right\|_F \leq \sum_{k=2}^{\infty} \left\| |G|^k \right\|_F \leq \frac{\|G\|_F^2}{1-\|G\|_F} < 1.$$

Analogously,

$$|w_i| \leq \|W\|_\infty \leq \frac{\|G\|_\infty^2}{1-\|G\|_\infty} < 1.$$

Thus we can write $1 + w_i = (1+d_i')^2$, with $|d_i'| \leq |w_i|$ for all i. Then

$$\bar{D} = J(I+W) = (I+D')J(I+D')$$

and (20) is proved.

We define $L' \equiv \bar{L} - I$ and combine (15), (18), and (20) to get

$$C = (I+L')(I+D')J(I+D')^T(I+L')^T.$$

Therefore

$$C = (I+\widetilde{E})J(I+\widetilde{E})^T, \text{ where } \|\widetilde{E}\|_F \leq \frac{1}{\sqrt{2}} \cdot \frac{\|G\|_F}{1-\|G\|_F} + \frac{\|G\|_F^2}{1-\|G\|_F} + \frac{1}{\sqrt{2}} \cdot \frac{\|G\|_F^3}{(1-\|G\|_F)^2}.$$

Note that $I + \widetilde{E} = (I+L')(I+D')$ is lower triangular. Taking into account that $\|G\|_F \leq \frac{1}{5}$ by (14), we can simplify the bound on $\|\widetilde{E}\|_F$ as follows

$$\begin{aligned} \|\widetilde{E}\|_F &\leq \frac{\|G\|_F}{1-\|G\|_F} \left(\frac{1}{\sqrt{2}} + \|G\|_F + \frac{1}{\sqrt{2}} \cdot \frac{\|G\|_F^2}{1-\|G\|_F} \right) \\ &\leq \frac{\|G\|_F}{1-\|G\|_F} \left(\frac{1}{\sqrt{2}} + \frac{1}{5} + \frac{1}{\sqrt{2}} \cdot \frac{1/5^2}{1-(1/5)} \right) \\ &< \frac{\|G\|_F}{1-\|G\|_F}. \end{aligned} \qquad (21)$$

For the ∞-norm, we use (19) instead of (18) to get

$$\begin{aligned}\|\widetilde{E}\|_\infty &\le \frac{\|G\|_\infty}{1-\|G\|_\infty}\left(1+\|G\|_\infty+\frac{\|G\|_\infty^2}{1-\|G\|_\infty}\right)\\ &\le \frac{\|G\|_\infty}{1-\|G\|_\infty}\left(1+\frac{1}{5}+\frac{1/5^2}{1-(1/5)}\right)\\ &= \frac{5}{4}\,\frac{\|G\|_\infty}{1-\|G\|_\infty}. \end{aligned} \tag{22}$$

We are now in a position to finish the proof of the Theorem. We write

$$\begin{aligned} A = D_A C D_A &= D_A(I+\widetilde{E})D_A^{-1}D_A J D_A D_A^{-1}(I+\widetilde{E})^T D_A\\ &= (I+D_A\widetilde{E}D_A^{-1})\,\mathrm{diag}(a_{11},\dots,a_{nn})\,(I+D_A\widetilde{E}D_A^{-1})^T, \end{aligned}$$

define $E \equiv D_A\widetilde{E}D_A^{-1}$, and use (14) and (21) to prove the second claim for the Frobenius norm. For the second claim in the ∞-norm, use (14) and (22). In addition $|e_{ij}| \le |\widetilde{e}_{ij}|$, since $\widetilde{E}$ is lower triangular, and $|a_{11}| \ge \cdots \ge |a_{nn}|$. Therefore $\|E\|_F \le \|\widetilde{E}\|_F < 1$. To prove the first claim, take $I+F = (I+E)^{-1} = \sum_{k=0}^{\infty}(-E)^k$, and write

$$\|F\|_F \le \frac{\|E\|_F}{1-\|E\|_F} \le \frac{\frac{n\delta}{1-n\delta}}{1-\frac{n\delta}{1-n\delta}} = \frac{n\delta}{1-2n\delta}.$$

□

Finally, we prove the main result in this section—that there is no severe cancellation in computing the diagonal entries of a scaled diagonally dominant matrix from its RRD.

Theorem 5 *Let $X \in \mathbb{R}^{n\times n}$ be nonsingular and $D = \mathrm{diag}(d_1,\dots,d_n) \in \mathbb{R}^{n\times n}$ be diagonal and nonsingular. If the matrix $A \equiv XDX^T$ satisfies $a_{ii} \ne 0$ for all i, and*

$$\frac{|a_{ij}|}{\sqrt{|a_{ii}a_{jj}|}} \le \delta, \quad \textit{for all } i \ne j,$$

where $\delta \le \frac{1}{5n}$, then

$$\frac{\sum_{k=1}^{n} x_{ik}^2|d_k|}{|a_{ii}|} \le \frac{\kappa(X)}{1-2n\delta}\left(1+\frac{3n^2\delta}{1-n\delta}\right), \qquad i=1,\dots,n.$$

Proof The result is invariant under permutations PAP^T of A, so we assume $|a_{11}| \ge \cdots \ge |a_{nn}|$. We apply Theorem 4-part 2 to $A = XDX^T$ to write

$$\mathrm{diag}(a_{11},\dots,a_{nn}) = (I+E)^{-1}XDX^T\,(I+E)^{-T} \equiv \widetilde{X}D\widetilde{X}^T,$$

where $\widetilde{X} \equiv (I+E)^{-1}X$. We apply Theorem 3 to the diagonal matrix $\widetilde{X}D\widetilde{X}^T$ to obtain

$$\frac{\sum_{k=1}^{n}\widetilde{x}_{ik}^2|d_k|}{|a_{ii}|} \le \kappa(\widetilde{X}) \ \text{ for } i=1,\dots,n. \tag{23}$$

For all i,k,

$$x_{ik} = \widetilde{x}_{ik} + \sum_{j=1}^{n} e_{ij}\widetilde{x}_{jk}, \quad \text{and} \quad |x_{ik}| \le |\widetilde{x}_{ik}| + \sum_{j=1}^{n}|e_{ij}||\widetilde{x}_{jk}|.$$

However from (23)

$$|\widetilde{x}_{ik}| \le \sqrt{\frac{|a_{ii}|}{|d_k|}}\sqrt{\kappa(\widetilde{X})} \text{ for all } i,k.$$

Therefore for all i,k

$$\begin{aligned}
x_{ik}^2 &\le \widetilde{x}_{ik}^2 + 2\sum_{j=1}^{n} |e_{ij}||\widetilde{x}_{ik}||\widetilde{x}_{jk}| + \left(\sum_{j=1}^{n}|e_{ij}||\widetilde{x}_{jk}|\right)^2 \\
&\le \widetilde{x}_{ik}^2 + \frac{\kappa(\widetilde{X})}{|d_k|}\left(2\sum_{j=1}^{n}|e_{ij}|\sqrt{|a_{ii}a_{jj}|} + \left(\sum_{j=1}^{n}|e_{ij}|\sqrt{|a_{jj}|}\right)^2\right) \\
&= \widetilde{x}_{ik}^2 + \frac{\kappa(\widetilde{X})|a_{ii}|}{|d_k|}\left(2\sum_{j=1}^{n}|e_{ij}|\sqrt{\frac{|a_{jj}|}{|a_{ii}|}} + \left(\sum_{j=1}^{n}|e_{ij}|\sqrt{\frac{|a_{jj}|}{|a_{ii}|}}\right)^2\right).
\end{aligned}$$

Observe that with the notation of Theorem 4 we have that $(D_A^{-1}ED_A)_{ij} = e_{ij}\sqrt{\frac{|a_{jj}|}{|a_{ii}|}}$ and that $\|D_A^{-1}ED_A\|_\infty \le \frac{5}{4}\delta_n$, where $\delta_n \equiv \frac{n\delta}{1-n\delta}$. Therefore for all i,k

$$\begin{aligned}
x_{ik}^2 &\le \widetilde{x}_{ik}^2 + \frac{\kappa(\widetilde{X})|a_{ii}|}{|d_k|}\left(2\|D_A^{-1}ED_A\|_\infty + \|D_A^{-1}ED_A\|_\infty^2\right) \\
&\le \widetilde{x}_{ik}^2 + \frac{\kappa(\widetilde{X})|a_{ii}|}{|d_k|}\,\delta_n\left(\frac{5}{2} + \frac{25}{16}\delta_n\right) \\
&\le \widetilde{x}_{ik}^2 + \frac{\kappa(\widetilde{X})|a_{ii}|}{|d_k|}\,\delta_n\left(\frac{5}{2} + \frac{25}{16}\frac{1/5}{1-1/5}\right) \\
&< \widetilde{x}_{ik}^2 + \frac{\kappa(\widetilde{X})|a_{ii}|}{|d_k|}\,3\,\delta_n,
\end{aligned}$$

which combined with (23) implies

$$\frac{\sum_{k=1}^{n} x_{ik}^2|d_k|}{|a_{ii}|} \le \kappa(\widetilde{X})\,(1+3\,\delta_n\,n)\,. \tag{24}$$

The result now follows by observing that $\widetilde{X} \equiv (I+E)^{-1}X$ and thus

$$\kappa(\widetilde{X}) \le \kappa(X)\|I+E\|_2\|(I+E)^{-1}\|_2 \le \kappa(X)\frac{1+\|E\|_2}{1-\|E\|_2} \le \frac{\kappa(X)}{1-2n\delta},$$

where we used $\|E\|_2 \le \|E\|_F \le \|D_A^{-1}ED_A\|_F \le \delta_n$ since, from Theorem 4, E is lower triangular. □

5 Rounding error analysis of the implicit Jacobi algorithm

In the rounding error analysis of Algorithm 1 we use the conventional error model for floating point arithmetic [29, section 2.2]:

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

where a and b are real floating point numbers, $\odot \in \{+,-,\times,/\}$, and $|\delta| \leq \varepsilon$, with ε the machine precision. We assume that this holds also for the square root operation and that neither overflow nor underflow occurs. We also use the following notation introduced in [29, Chapter 3]: θ_q is any number such that

$$|\theta_q| \leq \frac{q\varepsilon}{1-q\varepsilon} \equiv \gamma_q.$$

The constants appearing in the error bounds are not important unless they become too large. Thus, we use a big-O notation that shows the dimensional dependence for the bounds. More precisely, if $p(n)$ is a moderately increasing function in n

$$h(\varepsilon) = O(p(n)\,\varepsilon) \quad \text{means} \quad |h(\varepsilon)| \leq \alpha\,|p(n)|\,\varepsilon,$$

where α denotes a small integer constant that does not depend on the dimension n of the problem. We also use sometimes the following notation introduced in [29, p. 68]

$$\widetilde{\gamma}_q = \frac{\alpha q\varepsilon}{1-\alpha q\varepsilon}.$$

We warn the reader that for making the notation as simple as possible, $fl(expression)$ will denote the computed value in floating point arithmetic of any *expression*, where, in turn, every variable appearing in *expression* has to be computed in floating point arithmetic.

5.1 Rounding errors in the stopping criterion of Algorithm 1

Note that the stopping criterion in Algorithm 1 involves entries of the matrix GJG^T that have to be computed from the factors J and G, where G is the last iterate of the implicit Jacobi algorithm. Therefore, the errors introduced by the stopping criterion have to be carefully analyzed. This analysis starts with Lemma 2 that shows that if the stopping criterion is satisfied in floating point arithmetic then the exact matrix GJG^T is scaled diagonally dominant with a slightly different constant. In Lemma 2 we consider arbitrary thresholds τ_1 and τ_2, but remember that, according to Theorem 5, τ_2 should be, more or less, $\kappa(X)$. In Algorithm 1 we have used use $\tau_2 = 2\,\widehat{\kappa}(X)$. Recall that the entries of GJG^T in Algorithm 1 are computed with formula (4).

Lemma 2 *Let $G \in \mathbb{R}^{n\times n}$ be a matrix of floating point numbers and $J = \operatorname{diag}(s_1,\ldots,s_n) \in \mathbb{R}^{n\times n}$ be a diagonal matrix whose entries are $s_i = \pm 1$. Let $A = GJG^T$ as in* (4). *If*

$$fl\left(\frac{|a_{ij}|}{\sqrt{|a_{ii}a_{jj}|}}\right) \leq \tau_1 \quad \textit{for all } i \neq j, \tag{25}$$

$$fl\left(\frac{\sum_{k=1}^{n} g_{ik}^2}{|a_{ii}|}\right) \leq \tau_2 \quad \textit{for all } i, \tag{26}$$

and $\tau_2\gamma_{n+1} < \frac{1}{4}$ (observe that $\tau_2 > 1$), then

1. $fl\,(a_{ii}) = a_{ii}\,(1+\phi_i)$ *with* $|\phi_i| \leq \dfrac{\gamma_n\tau_2}{1-2\,\tau_2\gamma_{n+1}}$ *for all* i.
2. $$\frac{|a_{ij}|}{\sqrt{|a_{ii}a_{jj}|}} \leq \tau_1\left(1+\frac{\gamma_n\tau_2}{1-2\,\tau_2\gamma_{n+1}}\right) + \frac{\gamma_{n+3}\tau_2}{1-2\,\tau_2\gamma_{n+1}}, \quad \textit{for all } i \neq j. \tag{27}$$

Proof The proof is standard rounding error analysis. We simply sketch it. Define $B \equiv GG^T$. Then $a_{ii} = \sum_{k=1}^n g_{ik}^2 s_k$, $b_{ii} = \sum_{k=1}^n g_{ik}^2$ and

$$fl(a_{ii}) = \sum_{k=1}^{n} g_{ik}^2 s_k (1+\theta_n^{(k)}) = a_{ii} + \theta_n b_{ii}. \tag{28}$$

From (26)

$$\frac{b_{ii} + \theta_{n+1} b_{ii}}{|a_{ii} + \theta_n b_{ii}|} \le \tau_2, \quad \text{so,} \quad \frac{b_{ii}}{|a_{ii}|} \le \frac{\tau_2}{1 - 2\,\tau_2 \gamma_{n+1}}, \quad \text{for all } i. \tag{29}$$

The first claim follows by combining (29) with (28). To prove the second claim, write

$$fl\left(\sqrt{|a_{ii}a_{jj}|}\right) = \sqrt{|a_{ii}a_{jj}|}\,(1+\phi_{ij})(1+\delta_1)(1+\delta_2),$$

where $|\phi_{ij}| \le \gamma_n \tau_2 / (1 - 2\,\tau_2 \gamma_{n+1})$ and $|\delta_1|, |\delta_2| \le \varepsilon$. Then (25) implies

$$\frac{|a_{ij} + \theta_{n+3} \sum_{k=1}^n |g_{ik}||g_{jk}||}{\sqrt{|a_{ii}a_{jj}|}\,(1+\phi_{ij})} \le \tau_1,$$

and

$$\frac{|a_{ij}|}{\sqrt{|a_{ii}a_{jj}|}} \le \tau_1\,(1+\phi_{ij}) + \gamma_{n+3} \frac{\sum_{k=1}^n |g_{ik}||g_{jk}|}{\sqrt{|a_{ii}a_{jj}|}}.$$

Finally, from Cauchy-Schwarz $\sum_{k=1}^n |g_{ik}||g_{jk}| \le \sqrt{b_{ii}b_{jj}}$, and the result follows from (29). □

The first important consequence of Lemma 2 is that one should not take $\tau_1 < \varepsilon \tau_2$ in the stopping criterion given by (25) and (26), because this stopping criterion implies the bound (27) on the matrix A, and the right hand side in (27) is larger than $\tau_1 + (n+3)\varepsilon\tau_2$. Therefore trying to make the matrix A more scaled diagonally dominant by choosing a very small threshold τ_1 in (25) has a marginal effect in the matrix. We have used in Algorithm 1: $\tau_2 = 2\,\widehat{\kappa}(X)$, where $\widehat{\kappa}(X)$ is the computed estimation of $\kappa(X)$, and $\tau_1 = \varepsilon \max\{n, \widehat{\kappa}(X)\}$ with good results.

The second important consequence of Lemma 2 is that it can be combined with Theorem 4 to prove that the stopping criterion causes a small multiplicative backward error in the RRD. From now on, we use big-O notation for the error bounds, because the precise constants may be complicated and are of no interest to us.

Lemma 3 *With the same notation and assumptions as in Lemma 2, suppose, in addition, that* $\tau_1 = O(\varepsilon)$. *Then*

$$\operatorname{diag}(fl(a_{11}), \ldots, fl(a_{nn})) = (I+F)GJG^T(I+F)^T,$$

where $\|F\|_F = O(n\tau_1 + n^2 \varepsilon \tau_2)$.

Proof From Lemma 2 and Theorem 4,

$$\operatorname{diag}(a_{11}, \ldots, a_{nn}) = (I+\widetilde{F})GJG^T(I+\widetilde{F})^T \quad \text{with} \quad \|\widetilde{F}\|_F = O(n\tau_1 + n^2\varepsilon\tau_2).$$

Define α_i from $1+\alpha_i = \sqrt{1+\phi_i}$ and let $E \equiv \operatorname{diag}(\alpha_1, \ldots, \alpha_n)$. Since $|\alpha_i| \le |\phi_i| = O(n\varepsilon\tau_2)$, $\|E\|_F = O(n^{3/2}\varepsilon\tau_2)$. Then

$$\begin{aligned}\operatorname{diag}(fl(a_{11}), \ldots, fl(a_{nn})) &= (I+E)\operatorname{diag}(a_{11}, \ldots, a_{nn})(I+E)^T \\ &= (I+E)(I+\widetilde{F})GJG^T(I+\widetilde{F})^T(I+E)^T\end{aligned}$$

and the result follows by defining F from $I+F = (I+E)(I+\widetilde{F})$. □

5.2 Multiplicative backward error analysis of Algorithm 1

We present in this section a detailed multiplicative backward error analysis for Algorithm 1. The multiplicative backward error bound in Theorem 6 below can be easily combined with Theorems 1 and 2 to prove that Algorithm 1 computes the eigenvalues and eigenvectors of an RRD XDX^T with errors (7). We start with the technical Lemma 4 that we use in the proof of the main Theorem 6. This lemma is essentially known (see [13, p. 251], [29, pp. 367, 360]), our contribution is simply to bound the Frobenius norm of the backward error in terms of the spectral norm of the matrix without any dimensional penalty.

Lemma 4 *Let $A \in \mathbb{R}^{n\times n}$, $\bar{D} \in \mathbb{R}^{n\times n}$ be any diagonal matrix with positive entries, $\widehat{R}_1, \ldots, \widehat{R}_q$ be computed Jacobi rotations, and $R_1, \ldots, R_q$ be exact Jacobi rotations. Assume that for each rotation $\widehat{R}_k$ the computed cosine, $\hat{c}$, and the computed sine, $\hat{s}$, satisfy*

$$\hat{c} = c\,(1+\theta_5) \quad \textit{and} \quad \hat{s} = s\,(1+\theta_5'), \tag{30}$$

where c and s are the exact cosine and sine corresponding to R_k. Then

$$fl(\widehat{R}_q \cdots \widehat{R}_1 A) = R_q \cdots R_1\,(A+F) \quad \textit{where} \quad \|F\bar{D}\|_F \leq \frac{\alpha q \varepsilon}{1-\alpha q \varepsilon}\|A\bar{D}\|_2,$$

and α denotes a small integer constant.

Remark 1 The constant q, i.e., the number of Jacobi rotations, in the error bound in Lemma 4 is pessimistic. For instance, if the Jacobi rotations correspond to a whole sweep then $q = n(n-1)/2$, but in the error bound one can put $2n-3$ if the notion of disjoint Jacobi rotations is used [24] (see also [18, Prop. 3.5]). We will express the rest of our error bounds in terms of the numbers of Jacobi rotations applied until the stopping criterion is satisfied, but the right quantity is the number of sets of disjoint Jacobi rotations that are applied. However, it is difficult to know exactly this number, specially in the last sweeps where just a few rotations may be applied.

Proof of Lemma 4 This is a standard error analysis. We simply sketch the proof. Let us study the application of one rotation $fl(\widehat{R}_1 A)$. It is well known (see [13, Lemma 3.1] or [29, Lemma 19.9]) that $fl(\widehat{R}_1 A) = R_1(A+F_1)$ with $\|F_1(:,k)\|_2 \leq \widetilde{\gamma}_1 \|A(:,k)\|_2$ for all k. But if R_1 is an $R(i,j,c,s)$ rotation then operations are only performed on rows i and j of A, so $F_1(l,:) = 0$ for $l \neq i$ and $l \neq j$, and we have the stronger bound $\|F_1(:,k)\|_2 = \|F_1([i,j],k)\|_2 \leq \widetilde{\gamma}_1 \|A([i,j],k)\|_2$. Therefore

$$\left\|\bar{d}_{kk}F_1(:,k)\right\|_2 \leq \widetilde{\gamma}_1 \left\|\bar{d}_{kk}A([i,j],k)\right\|_2 = \widetilde{\gamma}_1 \,\|(A\bar{D})\,([i,j],k)\|_2\,,$$

and

$$\|F_1\bar{D}\|_F \leq \widetilde{\gamma}_1 \,\|(A\bar{D})\,([i,j],:)\|_F \leq \widetilde{\gamma}_1 \sqrt{2}\,\|(A\bar{D})\,([i,j],:)\|_2 \leq \widetilde{\gamma}_1 \sqrt{2}\,\|A\bar{D}\|_2\,.$$

Finally, for one rotation

$$fl(\widehat{R}_1 A) = R_1(A+F_1) \quad \text{where} \quad \|F_1\bar{D}\|_F \leq \widetilde{\gamma}_1\,\|A\bar{D}\|_2\,, \tag{31}$$

where the factor $\sqrt{2}$ has been absorbed in the moderate constant used in the definition of $\widetilde{\gamma}_1$. Now the proof continues with an inductive argument. Denote $\widehat{A}_k = fl(\widehat{R}_k \cdots \widehat{R}_1 A)$ for $k = 1,2,\ldots,q$, and assume that the result holds for $\widehat{A}_{q-1}$. From (31), we get

$$\widehat{A}_q = R_q(\widehat{A}_{q-1} + \widetilde{F}_1) \quad \text{where} \quad \left\|\widetilde{F}_1\bar{D}\right\|_F \leq \widetilde{\gamma}_1 \left\|\widehat{A}_{q-1}\bar{D}\right\|_2. \tag{32}$$

Then,

$$\widehat{A}_q = R_q(R_{q-1}\cdots R_1(A+F_{q-1})+\widetilde{F}_1) \quad \text{where} \quad \left\|F_{q-1}\bar{D}\right\|_F \leq \widetilde{\gamma}_{q-1}\,\|A\bar{D}\|_2. \tag{33}$$

The result is obtained by combining (32) and (33), using that $\left\|\widehat{A}_{q-1}\bar{D}\right\|_2 = \left\|(A+F_{q-1})\bar{D}\right\|_2 \leq (1+\widetilde{\gamma}_{q-1})\,\|A\bar{D}\|_2$, and [29, Lemma 3.3]. □

Theorem 6 *If N_R Jacobi rotations are applied in Algorithm 1 until the stopping criterion is satisfied, $\widehat{\kappa}(X)\gamma_{n+1} < \frac{1}{8}$, and $\sqrt{n}\kappa(X)\gamma_2 < \frac{1}{2}$, then the computed matrix of eigenvalues, $\widehat{\Lambda} = \operatorname{diag}(\widehat{\lambda}_1,\ldots,\widehat{\lambda}_n)$, and the computed matrix of eigenvectors, $\widehat{U}$, are nearly the exact eigenvalue and eigenvector matrices of a small multiplicative perturbation of XDX^T. More precisely, there exists an exact orthogonal matrix $U \in \mathbb{R}^{n\times n}$ such that*

$$U\widehat{\Lambda}U^T = (I+E)XDX^T\,(I+E)^T, \tag{34}$$

with $\|E\|_F = O(\varepsilon\,(n^2\widehat{\kappa}(X)+N_R\kappa(X)))$ and $\|\widehat{U}-U\|_F = O(N_R\,\varepsilon)$.

Remark 2 Taking into account that the X factor of an RRD is a well-conditioned matrix any sensible way to estimate $\kappa(X)$ will produce an estimation such that $\kappa(X) \approx \widehat{\kappa}(X)$. Therefore, the bound above for E usually simplifies to

$$\|E\|_F \approx O(\varepsilon\,(n^2+N_R)\kappa(X)).$$

Proof of Theorem 6 Let $D \equiv \operatorname{diag}(d_1,\ldots,d_n)$. Then $|D|^{1/2} = \operatorname{diag}(\sqrt{|d_1|},\ldots,\sqrt{|d_n|})$. The computed version of $X|D|^{1/2}$ is

$$\widehat{G} = fl(X\,|D|^{1/2}), \quad \text{and satisfies} \quad \widehat{G} = \widetilde{X}\,|D|^{1/2} \ \text{ with } \ \widetilde{x}_{ij} = x_{ij}\,(1+\theta_2^{(ij)}). \tag{35}$$

Let $\widehat{G}_f = fl(\widehat{R}_{N_R}^T\cdots\widehat{R}_1^T\,\widehat{G})$ be the computed matrix after the N_R Jacobi rotations are applied. Recall that for each Jacobi rotation the cosine and sine are, respectively, $1/\sqrt{1+t^2}$ and $t/\sqrt{1+t^2}$, where the quantity t may be found for instance in [13, Section 5.3.5]. Even in the case that the computed $\hat{t}$ has a large relative error[5], the computed $\hat{c}$ and $\hat{s}$ satisfy (30) with respect to the following exact cosine $c = 1/\sqrt{1+\hat{t}^2}$ and sine $s = c\hat{t}$, and we can apply Lemma 4 to get

$$\widehat{G}_f = R_{N_R}^T\cdots R_1^T(\widehat{G}+F), \quad \text{where } \|F\,|D|^{-1/2}\|_F = O(N_R\varepsilon)\,\|\widehat{G}\,|D|^{-1/2}\|_2 = O(N_R\varepsilon)\,\|\widetilde{X}\|_2.$$

Then

$$\widehat{G}_f = R_{N_R}^T\cdots R_1^T(I+F')\widehat{G}, \quad \text{where } \|F'\|_F = O(N_R\varepsilon)\,\kappa(\widetilde{X}), \tag{36}$$

because $\|F'\|_F = \|F\widehat{G}^{-1}\|_F = \|F|D|^{-1/2}\widetilde{X}^{-1}\|_F \leq \|F|D|^{-1/2}\|_F\,\|\widetilde{X}^{-1}\|_2$.

The matrix $\widehat{G}_f J\widehat{G}_f^T$ satisfies the stopping criterion in finite arithmetic, and Lemma 3 can be applied with $\tau_1 = \varepsilon\,\max\{n,\widehat{\kappa}(X)\}$ and $\tau_2 = 2\widehat{\kappa}(X)$. We get

$$\widehat{\Lambda} = (I+\widetilde{F})\,\widehat{G}_f J\widehat{G}_f^T\,(I+\widetilde{F})^T, \quad \text{where } \|\widetilde{F}\|_F = O(n^2\varepsilon\widehat{\kappa}(X)). \tag{37}$$

[5] Large relative errors in $\hat{t}$ do not affect the error analysis and, so, do not affect the accuracy of the eigenvalues and eigenvectors. However, it should be remarked that they may affect the rate of convergence in floating point arithmetic and make the algorithm slow, especially at the beginning of the process.

Define the exact orthogonal matrix $U^T = R_{N_R}^T \cdots R_1^T$, and combine (36), (37), and (35) to obtain

$$\begin{aligned}\widehat{\Lambda} &= U^T(I+U\widetilde{F}U^T)(I+F')\widehat{G}J\widehat{G}^T(I+F')^T(I+U\widetilde{F}U^T)^T U \\ &= U^T(I+U\widetilde{F}U^T)(I+F')\widetilde{X}D\widetilde{X}^T(I+F')^T(I+U\widetilde{F}U^T)^T U. \qquad (38)\end{aligned}$$

Note that (35) implies that $\widetilde{X} = X + E_X = (I + E_X X^{-1})X$, with $\|E_X X^{-1}\|_F \le \|E_X\|_F \, \|X^{-1}\|_2 \le \gamma_2\sqrt{n}\kappa(X)$. If we define $I + E = (I + U\widetilde{F}U^T)(I+F')(I+E_X X^{-1})$ and use (38), then

$$U\widehat{\Lambda}U^T = (I+E)XDX^T(I+E)^T, \quad \text{where } \|E\|_F = O(\varepsilon(n^2\widehat{\kappa}(X) + N_R\kappa(\widetilde{X}) + \sqrt{n}\kappa(X))).$$

To obtain (34), it remains to relate $\kappa(\widetilde{X})$ and $\kappa(X)$. From $\widetilde{X} = (I + E_X X^{-1})X$,

$$\kappa(\widetilde{X}) \le \kappa(X)\frac{1 + \|E_X X^{-1}\|_2}{1 - \|E_X X^{-1}\|_2} \le \kappa(X)\frac{1+\gamma_2\sqrt{n}\kappa(X)}{1-\gamma_2\sqrt{n}\kappa(X)} \le 3\kappa(X).$$

This implies (34) under the mild assumption $N_R > \sqrt{n}$.

We still have to prove that $\|\widehat{U} - U\|_F = O(N_R\varepsilon)$. For this purpose, we use Lemma 4 with $\bar{D} = I$ and $A = I$ to prove that

$$\widehat{U} = fl(\widehat{R}_1 \cdots \widehat{R}_{N_R}) = (I + E_U)U, \quad \text{where } \|E_U\|_F = O(N_R\varepsilon)\|I\|_2 = O(N_R\varepsilon).$$

□

6 Singular RRDs: RRDs with rectangular factors

So far we have considered RRDs $A = XDX^T$ with square and nonsingular X and D, which excludes singular matrices A. If we only insist on X being nonsingular, then any zero eigenvalues of A will be explicitly revealed as zero entries on the diagonal of D. If $d_1, \ldots, d_r$, $r \le n$, are the nonzero diagonal entries of D, then for $\bar{D} = \operatorname{diag}(d_1, \ldots, d_r)$ and $\bar{X} = X(:, 1:r)$,

$$A = XDX^T = \bar{X}\bar{D}\bar{X}^T,$$

which leads us to consider rectangular RRDs. Computing the QR factorization of $\bar{X}$ is all that it takes to reduce the computation to Algorithm 1.

Algorithm 2 Given $X \in \mathbb{R}^{n\times r}$, $n > r$, of full column rank and $D \in \mathbb{R}^{r\times r}$ diagonal and nonsingular, this algorithm computes the eigenvalues $\lambda_1, \ldots, \lambda_n$ of $A = XDX^T$ and an orthogonal eigenvector matrix $U \in \mathbb{R}^{n\times n}$.

1. Let $Q\begin{bmatrix} R \\ 0 \end{bmatrix} = X$ be the QR factorization of X ($Q \in \mathbb{R}^{n\times n}, R \in \mathbb{R}^{r\times r}$) computed with Householder reflections [29, Ch. 19].
2. Let $\lambda_1, \ldots, \lambda_r$ and $U_R \in \mathbb{R}^{r\times r}$ be the output of Algorithm 1 applied on R and D.
3. Then $\lambda_1, \ldots, \lambda_r, 0, \ldots, 0$ are the n eigenvalues of A ($n - r$ zeros).
4. $U(:, 1:r) = Q(:, 1:r)U_R$.
5. $U(:, r+1:n) = Q(:, r+1:n)$.

The mathematical explanation for Algorithm 2 follows from the following block manipulation. If $\Lambda_R \equiv \operatorname{diag}(\lambda_1,\ldots,\lambda_r)$, then

$$A = Q\begin{bmatrix} RDR^T & 0 \\ 0 & 0\end{bmatrix} Q^T = Q\begin{bmatrix} U_R\Lambda_R U_R^T & 0 \\ 0 & 0\end{bmatrix} Q^T = Q\begin{bmatrix} U_R & 0 \\ 0 & I\end{bmatrix}\begin{bmatrix} \Lambda_R & 0 \\ 0 & 0\end{bmatrix}\begin{bmatrix} U_R^T & 0 \\ 0 & I\end{bmatrix} Q^T.$$

Next, we show that Algorithm 2 introduces small multiplicative backward errors of order $O(\varepsilon\,\kappa(X))$ and thus it computes the eigenvalues and eigenvectors of A to high relative accuracy.

Theorem 7 *Let N_R be the number of Jacobi rotations applied in step 2 of Algorithm 2. Let $\widehat{R}$ be the R-factor computed in step 1, and $\widehat{\kappa}(\widehat{R})$ be the computed estimation of the condition number of $\widehat{R}$ used in the stopping criterion of step 2. If $\widehat{\kappa}(\widehat{R})\gamma_{r+1} < \frac{1}{8}$ and $\sqrt{r}\,\kappa(X)\,\widetilde{\gamma}_{nr} < \frac{1}{2}$, then Algorithm 2 computes a matrix of eigenvalues, $\widehat{\Lambda} = \operatorname{diag}(\widehat{\lambda}_1,\ldots,\widehat{\lambda}_r,0,\ldots,0) \in \mathbb{R}^{n\times n}$, and a matrix of eigenvectors, $\widehat{U} \in \mathbb{R}^{n\times n}$, that are nearly the exact eigenvalue and eigenvector matrices of a small multiplicative perturbation of XDX^T. More precisely, there exists an exact orthogonal matrix $U \in \mathbb{R}^{n\times n}$ such that*

$$U\widehat{\Lambda}U^T = (I+E)XDX^T\,(I+E)^T, \tag{39}$$

with $\|E\|_F = O(\varepsilon\,(r^2\widehat{\kappa}(\widehat{R}) + \max\{N_R, r^{3/2}n\}\,\kappa(X)))$ and $\|\widehat{U}-U\|_F = O(\varepsilon\max\{n^{3/2}\,r, N_R\})$.

Remark 3 As in Remark 2, if the X factor is well-conditioned then $\kappa(X) \approx \widehat{\kappa}(\widehat{R})$.

Proof of Theorem 7 According to Theorem 19.4 and equation (19.13) in [29], there exists an exact orthogonal matrix Q such that the computed factors $\widehat{Q}$ and $\widehat{R}$ in step 1 of Algorithm 2 satisfy

$$X + \Delta X = Q\begin{bmatrix}\widehat{R} \\ 0\end{bmatrix}, \quad \text{where } \|\Delta X\|_F \le \widetilde{\gamma}_{nr}\,\|X\|_F, \text{ and } \|\widehat{Q}-Q\|_F \le \sqrt{n}\,\widetilde{\gamma}_{nr}. \tag{40}$$

Therefore, $X+\Delta X = (I+\Delta X X^{\dagger})X$, where $X^{\dagger}$ is the Moore-Penrose pseudoinverse of X, and

$$X + \Delta X = (I+F_X)X, \quad \text{where } \|F_X\|_F \le \sqrt{r}\,\widetilde{\gamma}_{nr}\,\kappa(X).$$

Then,

$$(I+F_X)XDX^T(I+F_X)^T = Q\begin{bmatrix}\widehat{R} \\ 0\end{bmatrix} D\begin{bmatrix}\widehat{R}^T & 0\end{bmatrix} Q^T = Q\begin{bmatrix}\widehat{R}D\widehat{R}^T & 0 \\ 0 & 0\end{bmatrix} Q^T.$$

If $\widehat{\Lambda}_R$ is the computed eigenvalue matrix of $\widehat{R}D\widehat{R}^T$ in step 2 of Algorithm 2, then Theorem 6 implies that there exists an orthogonal matrix $U_R \in \mathbb{R}^{r\times r}$ such that

$$(I+F_X)XDX^T(I+F_X)^T = Q\begin{bmatrix}(I+E_R)^{-1} & 0 \\ 0 & I\end{bmatrix}\begin{bmatrix}U_R\widehat{\Lambda}_R U_R^T & 0 \\ 0 & 0\end{bmatrix}\begin{bmatrix}(I+E_R)^{-T} & 0 \\ 0 & I\end{bmatrix} Q^T, \tag{41}$$

with $\|E_R\|_F = O(\varepsilon\,(r^2\widehat{\kappa}(\widehat{R}) + N_R\,\kappa(\widehat{R})))$. In this bound we can replace $\kappa(\widehat{R})$ by $\kappa(X)$ since[6]

$$\kappa(\widehat{R}) = \kappa((I+F_X)X) \le \kappa(X)\frac{1+\|F_X\|_2}{1-\|F_X\|_2} \le \kappa(X)\frac{1+\sqrt{r}\,\widetilde{\gamma}_{nr}\,\kappa(X)}{1-\sqrt{r}\,\widetilde{\gamma}_{nr}\,\kappa(X)} \le 3\kappa(X). \tag{42}$$

[6] Note that X is rectangular, therefore to get (42) we need to use [30, Theorem 3.3.16] which implies $\|(I+F_X)^{-1}\|_2^{-1}\sigma_i(X) \le \sigma_i((I+F_X)X) \le \|I+F_X\|_2\sigma_i(X)$, for $i = 1,\ldots,r$, where the σ_is denote singular values.

Define

$$I + E = Q \begin{bmatrix} I + E_R & 0 \\ 0 & I \end{bmatrix} Q^T (I + F_X),$$

note that $\|E\|_F = O(\varepsilon\,(r^2\,\widehat{\kappa}(\widehat{R}) + \max\{N_R, nr^{3/2}\}\kappa(X)))$, and use (41) to obtain

$$(I+E)XDX^T(I+E)^T = Q \begin{bmatrix} U_R & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \widehat{\Lambda}_R & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U_R^T & 0 \\ 0 & I \end{bmatrix} Q^T.$$

This is (39) with $U = Q\left[\begin{smallmatrix} U_R & 0 \\ 0 & I \end{smallmatrix}\right]$.

Next, we bound $\|\widehat{U} - U\|_F$. If $\widehat{U}_R$ is the eigenvector matrix computed in step 2 of Algorithm 2, then

$$\widehat{U} = \left[fl(\widehat{Q}(:,1:r)\,\widehat{U}_R) \quad \widehat{Q}(:,r+1:n) \right],$$

and

$$\|\widehat{U} - U\|_F = \sqrt{\|fl(\widehat{Q}(:,1:r)\,\widehat{U}_R) - Q(:,1:r)\,U_R\|_F^2 + \|\widehat{Q}(:,r+1:n) \ - Q(:,r+1:n)\|_F^2}.$$

From (40), we get $\|\widehat{Q}(:,r+1:n) \ - Q(:,r+1:n)\|_F \leq \sqrt{n}\,\widetilde{\gamma}_{nr}$. Taking into account that $\|\widehat{U}_R - U_R\|_F = O(N_R\varepsilon)$, $\|U_R\|_F = \|Q(:,1:r)\|_F = \sqrt{r}$, (40), and the standard error bound for matrix multiplication [29, eq. (3.13)], we can prove that

$$\begin{aligned} \|fl(\widehat{Q}(:,1:r)\,\widehat{U}_R) - Q(:,1:r)\,U_R\|_F &\leq \|fl(\widehat{Q}(:,1:r)\,\widehat{U}_R) - \widehat{Q}(:,1:r)\,\widehat{U}_R\|_F \\ &\quad + \|\widehat{Q}(:,1:r)\,\widehat{U}_R - Q(:,1:r)\,U_R\|_F \\ &= O(r^2\varepsilon) + O(N_R\varepsilon) + O(n^{3/2}r\varepsilon). \end{aligned}$$

Therefore, $\|\widehat{U} - U\|_F = O(\varepsilon \max\{n^{3/2}\,r, N_R\})$. □

7 The effect of errors in X and D

In this section we consider the situation where the factors X and D in the RRD $A = XDX^T$ carry some errors from previous computations. This is the typical scenario in practice where the RRD is not given, but rather computed in floating point arithmetic. It turns out that the factors X and D accurately determine the eigendecomposition of A. In other words it suffices that X be computed with a small relative norm error and the diagonal of D be computed with a small relative componentwise error.

Lemma 5 *Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix and $A = XDX^T$ be a factorization of A, where $X \in \mathbb{R}^{n\times r}$ has a full column rank and $D = \operatorname{diag}(d_1,\ldots,d_r) \in \mathbb{R}^{r\times r}$ is diagonal and nonsingular. Let $\widehat{X}$ and $\widehat{D} = \operatorname{diag}(\widehat{d}_1,\ldots,\widehat{d}_r)$ be perturbations of X and D, respectively, that satisfy*

$$\frac{\|\widehat{X} - X\|_2}{\|X\|_2} \leq \delta \quad \textit{and} \quad \frac{|\widehat{d}_i - d_i|}{|d_i|} \leq \delta \quad \textit{for } i = 1,\ldots,r, \tag{43}$$

where $\delta < 1$. Then

$$\widehat{X}\widehat{D}\widehat{X}^T = (I+F)A(I+F)^T,$$

with $\|F\|_2 \leq (2\delta + \delta^2)\kappa(X)$.

Proof Let $\widehat{d_i} = d_i(1+\mu_i)$ where $|\mu_i| \le \delta < 1$, $i = 1,2,\ldots,r$. Then $\widehat{d_i} = (1+\delta_i)d_i(1+\delta_i)$, where $1+\delta_i = \sqrt{1+\mu_i}$, and $|\delta_i| \le \delta$. Define $W = \mathrm{diag}(\delta_1,\ldots,\delta_r)$. Then

$$\widehat{D} = (I+W)D(I+W)^T,$$

where $\|W\|_2 \le \delta$. We write $\widehat{X} = X+E$, with $\|E\|_2 \le \delta\,\|X\|_2$, and denote by $X^\dagger$ the pseudoinverse of X. Then

$$\begin{aligned}\widehat{X}\widehat{D}\widehat{X}^T &= (I+EX^\dagger)X(I+W)D(I+W)^T X^T(I+EX^\dagger)^T \\ &= (I+EX^\dagger)(I+XWX^\dagger)XDX^T(I+XWX^\dagger)^T(I+EX^\dagger)^T.\end{aligned}$$

The result follows from defining F from $I+F = (I+EX^\dagger)(I+XWX^\dagger)$. □

We now combine Lemma 5 with Theorem 6 or 7 to yield the final multiplicative backward error result for the computed eigenvalues and eigenvectors of a symmetric matrix A whose RRD is computed accurately with error bounds as in (43). For simplicity, we assume that the computed condition numbers $\widehat{\kappa}(X)$ and $\widehat{\kappa}(\widehat{R})$ appearing in Theorems 6 and 7, respectively, are good approximations to $\kappa(X)$.

Theorem 8 *Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix and $A = XDX^T$ be a factorization of A, where $X \in \mathbb{R}^{n\times r}$ has full column rank and $D \in \mathbb{R}^{r\times r}$ is diagonal and nonsingular. Let $\widehat{X}$ and $\widehat{D}$ satisfy* (43)*, with $\delta = O(\varepsilon)$ such that $\delta\,\kappa(X) < \frac{1}{2}$. Let $\widehat{\Lambda}$ and $\widehat{U}$ be the eigenvalue and eigenvector matrices of $\widehat{X}\widehat{D}\widehat{X}^T$ computed by Algorithm 2 (or Algorithm 1 if $n = r$). Let N_R be the number of Jacobi rotations applied in step 2 of Algorithm 2. Then there exists an exact orthogonal matrix $U \in \mathbb{R}^{n\times n}$ such that*

$$U\widehat{\Lambda}U^T = (I+F)A(I+F)^T, \tag{44}$$

where $\|F\|_2 = O((\varepsilon\cdot\max\{N_R, r^{3/2}n\}+\delta)\cdot\kappa(X))$ and $\|\widehat{U}-U\|_F = O(\varepsilon\cdot\max\{n^{3/2}r, N_R\})$.

Proof From (39), $U\widehat{\Lambda}U^T = (I+E)\widehat{X}\widehat{D}\widehat{X}^T(I+E)^T$, and from Lemma 5,

$$U\widehat{\Lambda}U^T = (I+E)(I+E_F)A(I+E_F)^T(I+E)^T.$$

This is (44) with $I+F = (I+E)(I+E_F)$, where we have replaced $\kappa(\widehat{X})$ in the bound for $\|E\|_F$ by $\kappa(X)$, because $\widehat{X} = X+E_X = (I+E_XX^\dagger)X$ implies

$$\kappa(\widehat{X}) \le \kappa(X)\frac{1+\delta\kappa(X)}{1-\delta\kappa(X)} \le 3\,\kappa(X),$$

by using [30, Theorem 3.3.16]. □

8 QR factorization as preconditioner and other implementation details

We will see in the numerical tests presented in Section 9 that Algorithm 1 can be very slow for RRDs with certain distributions of eigenvalues. The same is true for Algorithm 2 since it uses Algorithm 1. This poor performance from the point of view of speed may compromise the practical use of Algorithms 1 and 2 despite of their high accuracy. We present in this section a simple modification of Algorithm 1 that has an extremely positive impact in speeding up this algorithm.[7] In addition, we mention at the end of this section some other ideas on how to decrease the computational cost of each Jacobi sweep. The implementation and error analysis of these ideas is postponed to future research, but they show that there are many possible ways of improving the run-time performance of the implicit Jacobi algorithm.

In [20, Section 2], the QR factorization with column pivoting has been used as a very efficient preconditioner to speed up the one-sided Jacobi algorithm for the SVD. On the other hand, in the case of positive definite D, Algorithm 1 essentially reduces to the one-sided Jacobi algorithm for the SVD, so it is natural to try also the QR factorization as a preconditioner of the new implicit Jacobi algorithm. The specific procedure is the following: let $G = X\sqrt{|D|}$, where $\sqrt{|D|} = \mathrm{diag}(\sqrt{|d_1|}, \ldots, \sqrt{|d_n|})$, be the matrix defined in Algorithm 1, and $G = QR\Pi$ be the QR factorization of G computed with the Businger-Golub column pivoting strategy [4], where Π is a permutation matrix. Then, with the notation of Algorithm 1,

$$A = XDX^T = GJG^T = QR\Pi J\Pi^T R^T Q^T = Q(RJ'R^T)Q^T,$$

where $J' = \Pi J \Pi^T$ is the permuted diagonal signature matrix. This means that the QR factorization with pivoting of G can be seen as an implicit preconditioning of A by orthogonal similarity via Q. Now, one uses the implicit Jacobi algorithm on $RJ'R^T$ by applying the Jacobi rotations on the left side of R. The formal algorithm for this process is Algorithm 3.

Algorithm 3 (**QR-Preconditioned Implicit cyclic-by-row Jacobi on $\mathbf{XDX^T}$**) Given $X \in \mathbb{R}^{n\times n}$ well conditioned and nonsingular, and $D = \mathrm{diag}(d_1, \ldots, d_n) \in \mathbb{R}^{n\times n}$ diagonal and nonsingular, this algorithm computes the eigenvalues $\lambda_1, \ldots, \lambda_n$ of $A = XDX^T$ and an orthogonal matrix $U \in \mathbb{R}^{n\times n}$ of eigenvectors to high relative accuracy.

```
$\widehat{\kappa}(X)$ is the computed estimation of $\kappa(X)$
$G = X\,\mathrm{diag}(\sqrt{|d_1|}, \ldots, \sqrt{|d_n|})$
$J = \mathrm{diag}(\mathrm{sign}(d_1), \ldots, \mathrm{sign}(d_n))$
$QR\Pi = G$ is the QR factorization of $G$ with column pivoting
$U = Q$
$J' = \Pi J \Pi^T$
repeat
    for $i = 1 : n-1$
        for $j = i+1 : n$
            compute $b_{ii}, b_{ij}, b_{jj}$ of $B = RJ'R^T$ and $T = \left[\begin{smallmatrix} c & -s \\ s & c \end{smallmatrix}\right]$, $c^2 + s^2 = 1$, such that

                $T^T \begin{bmatrix} b_{ii} & b_{ij} \\ b_{ij} & b_{jj} \end{bmatrix} T = \begin{bmatrix} \mu_1 & \\ & \mu_2 \end{bmatrix}$

            $R = R(i,j,c,s)^T R$
            $U = U R(i,j,c,s)$
        endfor
    endfor
until convergence $\left( \frac{|b_{ij}|}{\sqrt{|b_{ii}b_{jj}|}} \le \varepsilon \max\{n, \widehat{\kappa}(X)\}$ for all $i < j$ and $\frac{\sum_{k=1}^{n} r_{ik}^2}{|b_{ii}|} \le 2\widehat{\kappa}(X)$ for all $i \right)$
compute $\lambda_k = b_{kk}$ for $k = 1, 2, \ldots, n$.
```

[7] The preconditioning technique presented in this section was suggested by Z. Drmač for which we are very grateful.

The numerical experiments in Section 9 will show that the number of sweeps performed by Algorithm 3 may be much smaller than the number of sweeps of the unpreconditioned Algorithm 1. The reduction in the number of sweeps depends heavily on the distribution of the eigenvalues, and in some situations only a few sweeps are saved. A complete explanation of the behaviour of this preconditioner is not known, even in the positive definite case, but some insights may be found in [20]. Taking into account that the computational cost of one Jacobi sweep is comparable to the cost of the QR factorization,[8] the overhead cost of the QR factorization is paid off with just one saved sweep. This preconditioner extends trivially to the rectangular case considered in Algorithm 2 without any additional cost, since in Algorithm 2 a QR factorization is already computed. The only needed modification is to first compute $G = X\,\mathrm{diag}(\sqrt{|d_1|},\ldots,\sqrt{|d_r|})$, and then to compute the QR factorization with column pivoting of G.

A final consideration with respect to the QR preconditioner is that it preserves the high relative accuracy of the implicit Jacobi algorithm on RRDs. The reason is simply that the rounding error analysis presented in Theorem 6 remains valid except by some minor changes in the constants of the error bounds of $\|E\|_F$ and $\|\hat{U}-U\|_F$. To see this, note first that pivoting does not affect the analysis because we can assume that X and D are ordered in advance in such a way that permutations are not needed. Second, that the QR factorization of G consists of applying orthogonal transformations on the left of G, so producing a columnwise backward error that is independent of the column scaling (see [29, Lemma 19.3, Theorem 19.4], for Householder versions of this fact). This implies that a backward error relation similar to (36) can be proved, and the rest of the proof remains the same.

Apart from reducing the number of sweeps, it is essential to decrease the computational cost per step to get an efficient implementation of the implicit Jacobi algorithm. The first issue is to pick the side on which the transformations are performed. For instance, in Fortran, the updating step $R(i,j,c,s)^T R \to R$ in Algorithm 3 is much slower than $R^T R(i,j,c,s) \to R^T$ because the arrays are stored by columns. This can be addressed simply by rewriting the algorithm in terms of R^T. Another idea is to use the Rutishauser's formulas to update the diagonal entries [39] combined with keeping the diagonal entries in a separate vector (see [42, Section 3.3.1] for details on how this can be implemented in a one-sided Jacobi algorithm). This may save the computation of the diagonal entries b_{ii} and b_{jj} in each step. However Rutishauser's formulas may introduce large errors that can spoil the accuracy of the algorithm, so their use must be accompanied by safety tests to decide when they can be applied and by explicit updating of the diagonal entries at the end of each sweep. Finally, self-scaled Jacobi rotations [1] may be used to save $2n$ flops each time a Jacobi rotation is applied [42, Section 3.4.1].

Other interesting (and more complicated) ideas that can reduce the cost of the implicit Jacobi algorithm are the following. First, to compute the eigenvector matrix a posteriori by solving a linear system in the spirit of [19] instead of accumulating the Jacobi rotations. This is motivated by the fact that we know the original matrix G and the final G_f obtained when the stopping criterion is satisfied, and then one can solve for U in the system $U^T G = G_f$. The numerical orthogonality in floating point arithmetic of the matrix U so computed should be carefully analyzed, and U should be reorthogonalized if it is necessary. Second, it may be possible to use a triangular form of the factor G after the preconditioning by the QR factorization with column pivoting to design better pivoting strategies, as it was done in [21] for a Jacobi SVD algorithm. Finally, the use of block versions of the implicit

[8] In fact a BLAS 3 implementation of the QR factorization is faster than one Jacobi sweep.

Jacobi algorithm apparently would inherit the same accuracy properties and may make the algorithm much faster. See [26,27] for references on block Jacobi procedures.

9 Numerical tests

We have proved rigorously in Theorems 6 and 7 that the implicit standard Jacobi algorithm computes the eigenvalues and eigenvectors of an RRD XDX^T with small multiplicative backward errors. This is combined with Theorems 1 and 2 to show that the forward errors in the computed eigenvalues and eigenvectors are given by (7). In addition Algorithm 3, i.e., the QR-preconditioned version, has the same error bounds. Therefore, it is not surprising that extensive numerical tests performed on different types of RRDs have confirmed the high relative accuracy of Algorithms 1, 2 and 3. We present in this section some selected numerical tests on RRDs with extremely ill-conditioned diagonal factors, and compare the performance in number of Jacobi sweeps of the new implicit standard Jacobi algorithm, with and without preconditioning, with other algorithms existing in the literature that compute with high relative accuracy eigenvalues and eigenvectors of RRDs.

All of the numerical tests in this section have been performed in MATLAB 7.0 (R14) with $\varepsilon = 2^{-53}$. We will assume that the eigenvalues are decreasingly ordered, i.e., $\lambda_1 \geq \ldots \geq \lambda_n$, and we define

$$\text{relgap}_i = \min\left(\min_{j \neq i} \frac{|\lambda_j - \lambda_i|}{|\lambda_i|}, 1\right).$$

Test 1. We consider a 100×100 symmetric Cauchy matrix A with entries

$$a_{ij} = \frac{1}{x_i + x_j}, \tag{45}$$

with $x_i = (-1)^{i-1} + (i-1)2^{-40}$ for $i = 1, 2, \ldots, 100$. For this matrix $\kappa(A) = 7.8 \cdot 10^{73}$. A symmetric RRD $A = XDX^T$ is computed using Algorithm 1 in [14] with errors as in (43) with $\delta = O(n^{3/2}\varepsilon)$, and then Algorithm 1 or 3 can be applied on the factors to obtain the eigenvalues and eigenvectors. The computed eigenvalues, $\hat{\lambda}_i$, and eigenvectors, $\hat{v}_i$, are compared with the eigenvalues, λ_i, and the eigenvectors, v_i, computed by the MATLAB `eig` function with 100-decimal digit arithmetic. The maximum relative errors in the eigenvalues and eigenvectors computed by Algorithm 1 are

$$\max_i \frac{|\hat{\lambda}_i - \lambda_i|}{|\lambda_i|} = 3.3 \cdot 10^{-14}, \quad \text{and} \quad \max_i \|\hat{v}_i - v_i\|_2 = 1.9 \cdot 10^{-14},$$

while the maximum relative errors in the eigenvalues and eigenvectors computed by Algorithm 3 are

$$\max_i \frac{|\hat{\lambda}_i - \lambda_i|}{|\lambda_i|} = 4.7 \cdot 10^{-15}, \quad \text{and} \quad \max_i \|\hat{v}_i - v_i\|_2 = 4.7 \cdot 10^{-15}.$$

The errors are very satisfactory in both cases. The number of Jacobi sweeps performed by Algorithm 1 is 35, and by Algorithm 3 is 4. This is the first example that we show to illustrate how beneficial the QR-preconditioner may be. Other interesting data in this test are: $\kappa(X) = 30.5$, and $\min_i \text{relgap}_i = 0.62$. The relative error in the eigenvalues computed by the MATLAB `eig` function in standard IEEE double precision arithmetic was $\max_i \frac{|\hat{\lambda}_i - \lambda_i|}{|\lambda_i|} =$

$3.25 \cdot 10^{55}$. Note that the eigenvalues of $A = GJG^T$, where $G = X\,\mathrm{diag}(\sqrt{|d_1|}, \dots, \sqrt{|d_n|})$, are equal to the eigenvalues of the matrix pencil $G^T G - \lambda J$. We have also used the MATLAB `eig`$(G^T G, J)$ function to compute the eigenvalues of this pencil, and the error was $\max_i \frac{|\hat{\lambda}_i - \lambda_i|}{|\lambda_i|} = 5.18 \cdot 10^{32}$.

Test 2. In this test we consider again a 100×100 symmetric Cauchy matrix A with entries given by (45), and with $x_i = i - 0.5$ for $i = 1, 2, \dots, 99$ and $x_{100} = -99.5$, and proceed as in Test 1. Note that A is the Hilbert matrix except for the last row and last column which are modified to make the matrix indefinite. For this matrix $\kappa(A) = 3.5 \cdot 10^{147}$. The computed eigenvalues, $\hat{\lambda}_i$, and eigenvectors, $\hat{v}_i$, are compared with the eigenvalues, λ_i, and the eigenvectors, v_i, computed by the MATLAB `eig` function with 200-decimal digit arithmetic. The maximum relative errors in the eigenvalues and eigenvectors computed by Algorithm 1 are

$$\max_i \frac{|\hat{\lambda}_i - \lambda_i|}{|\lambda_i|} = 1.2 \cdot 10^{-13}, \quad \text{and} \quad \max_i \|\hat{v}_i - v_i\|_2 = 5.7 \cdot 10^{-14},$$

while the maximum relative errors in the eigenvalues and eigenvectors computed by Algorithm 3 are

$$\max_i \frac{|\hat{\lambda}_i - \lambda_i|}{|\lambda_i|} = 4.9 \cdot 10^{-15}, \quad \text{and} \quad \max_i \|\hat{v}_i - v_i\|_2 = 3.9 \cdot 10^{-14}.$$

The errors in Algorithm 3 are considerably smaller than the errors in Algorithm 1. This may be related to the fact that the number of Jacobi sweeps performed by Algorithm 3 is just 5, while the number of sweeps performed by Algorithm 1 is 55. Other interesting data in this test are: $\kappa(X) = 45.22$ and $\min_i \mathrm{relgap}_i = 0.4$. The relative error in the eigenvalues computed by the MATLAB `eig` function in standard IEEE double precision arithmetic was $\max_i \frac{|\hat{\lambda}_i - \lambda_i|}{|\lambda_i|} = 1.84 \cdot 10^{132}$.

The results in Tests 1 and 2 are very satisfactory from the point of view of accuracy, both for Algorithm 1 and 3. In addition, the number of Jacobi sweeps performed by Algorithm 3 is low, while Algorithm 1 performs too many sweeps. We will see in the next numerical experiments that this sweep comparison varies widely for different types of RRDs, although Algorithm 3 is always faster than Algorithm 1, and, therefore, QR-preconditioning is highly recommended. Moreover, we will compare in the next tests the number of sweeps of Algorithms 1 and 3 with the number of sweeps of other algorithms of Jacobi type for computing the eigenvalues and eigenvectors of symmetric indefinite RRDs with high relative accuracy. The algorithms we use are the following ones.

1. The implicit one sided J-orthogonal Jacobi algorithm (see [42, Algorithm 3.3.1] or [41, Algorithm 1]). This algorithm uses hyperbolic transformations applied on the right side of the matrix G defined in Algorithm 1. This fact implies that the error bounds for the computed eigenvalues and eigenvectors are not guaranteed to be small, but, in practice, it has never been observed a significant loss of accuracy.
2. The SSVD algorithm presented in [16, Algorithm 1]. The SSVD algorithm does not preserve the symmetry of the problem because it uses Algorithm 3.1 in [7] to compute the SVD of the RRD XDX^T. Step 3 of that algorithm applies the one sided Jacobi algorithm for the SVD [12] to a certain matrix W applying the rotations from the right. In our tests, we have computed first the QR factorization with the Businger-Golub column pivoting strategy of W^T [20]. In this way the algorithm is faster and, simultaneously, the error bounds are guaranteed to be small.

In the tests below we have checked the accuracy of the eigenvalues computed by Algorithms 1 and 3 through the relative errors with respect to those computed by the J-orthogonal and the SSVD algorithm. For the eigenvectors, we also compare with the J-orthogonal and the SSVD algorithm, and multiply the norm of the difference of the ith eigenvectors by relgap_i. This has to be $O(\varepsilon\kappa(X))$.

Test 3. In this test we study the behavior of the number of Jacobi sweeps performed by Algorithms 1 and 3 as the condition number of the diagonal factor D of an RRD increases. We consider random RRDs XDX^T, where $X \in \mathbb{R}^{100\times 100}$ and $D \in \mathbb{R}^{100\times 100}$ are generated by the MATLAB command `gallery('randsvd',...)` developed by N. Higham [28]. For all tested RRDs $\kappa(X) = 30$, and the matrices X are generated with geometrically distributed singular values (`MODE = 3` in `gallery('randsvd',...)`). In a first type of RRDs the diagonal factors D are also generated with geometrically distributed singular values, and, in addition, the signs of the diagonal entries are randomly selected. We consider the following values of $\kappa(D) = 10^{10:20:110}$, and for each value of $\kappa(D)$ five RRDs are generated. The average numbers of sweeps are presented in Table 1. We have observed in the matrices of this test a maximum relative difference between the eigenvalues computed by Algorithm 1 and those computed by the other algorithms equal to $4.8\cdot 10^{-14}$. The maximum norm of the difference between eigenvectors multiplied by the corresponding relative gap has been $2.8\cdot 10^{-14}$. In a second type of RRDs, we repeat the same experiment with the only modification that the option `MODE = 1` in `gallery('randsvd',...)` is used to generate the absolute values of the diagonal factors D, i.e., D has one large singular value equal to one and the other singular values equal to $1/\kappa(D)$. Note that this does not imply that there are 99 eigenvalues of XDX^T with the same absolute value, because we are multiplying by X. The average numbers of sweeps for these RRDs are presented in Table 2. We have observed in the matrices of Table 2 a maximum relative difference between the eigenvalues computed by Algorithm 1 and those computed by the other algorithms equal to $2.8\cdot 10^{-14}$. The maximum norm of the difference between eigenvectors multiplied by the corresponding relative gap has been $3.3\cdot 10^{-14}$.

The first conclusion to be drawn from this test is that the performance of Algorithm 1 depends heavily on the distribution of the eigenvalues, while this dependence is milder for the other three algorithms. The second conclusion is that Algorithm 3 is always faster than Algorithm 1 and, that it can be much faster if the absolute values of the eigenvalues are geometrically distributed[9]. The third conclusion is that Algorithm 3 is also considerably faster than the J-Orthogonal algorithm, specially again if the absolute values of the eigenvalues are geometrically distributed. Therefore, we consider that the QR-preconditioning in Algorithm 3 must be used in the new Implicit Jacobi algorithm. The last conclusion is that the number of sweeps of the nonsymmetric SSVD algorithm is slightly smaller than the number of sweeps of Algorithm 3, although the lack of symmetry of SSVD makes difficult a real comparison of the computational cost of both algorithms. In addition, SSVD can use, at present, the fast algorithm in [20,21] to compute the SVD of XDX^T, and then to be faster than Algorithm 3.

Test 4. In this test we study the behavior of the number of Jacobi sweeps performed by Algorithms 1 and 3 as the dimension of an RRD XDX^T increases for fixed condition numbers of the factors D and X. We have chosen $\kappa(D) = 10^{40}$ and $\kappa(X) = 100$. For all tested RRDs, the matrices X are randomly generated with geometrically distributed singular values. In a first type of RRDs the diagonal factors D are also randomly generated with ge-

[9] The order of magnitude of the eigenvalues of XDX^T and D is similar because X is well-conditioned.

Table 1 Average numbers of Jacobi sweeps for random 100×100 RRDs with geometrically distributed singular values for D (`MODE=3`) and $\kappa(X) = 30$.

$\kappa(D)$	Algor. 1	Algor. 3	J-orth	SSVD
10^{10}	16.2	6	9	5.2
10^{30}	25	5	9	4
10^{50}	31.2	4	9.2	4.2
10^{70}	34.2	4	9	3.2
10^{90}	39.6	4	9.2	3
10^{110}	42.6	3.8	9.2	3

Table 2 Average numbers of Jacobi sweeps for random 100×100 RRDs with the singular values of D generated with `MODE=1` and $\kappa(X) = 30$.

$\kappa(D)$	Algor. 1	Algor. 3	J-orth	SSVD
10^{10}	10.2	9	10.8	8
10^{30}	9.6	8.8	10.6	8
10^{50}	10.6	9	10.4	8.2
10^{70}	10.8	9	10.8	8
10^{90}	11	8.8	10.8	8
10^{110}	11	8.6	11	8

Table 3 Average numbers of Jacobi sweeps for random $n \times n$ RRDs with geometrically distributed singular values for D (`MODE=3`), $\kappa(D) = 10^{40}$, and $\kappa(X) = 100$.

n	Algor. 1	Algor. 3	J-orth	SSVD
100	28.8	4.6	10	4
500	46	6	11	5.6
1000	58.3	6	11	6
2000	69	7	11	7

ometrically distributed singular values, and, in addition, the signs of the diagonal entries are randomly selected. We consider $n \times n$ factors X and D for $n = 100, 500, 1000, 2000$. Five RRDs were generated for $n = 100$, five for $n = 500$, three for $n = 1000$, and two for $n = 2000$. The average numbers of sweeps are presented in Table 3. We have observed in the matrices of this test a maximum relative difference between the eigenvalues computed by Algorithm 1 and those computed by the other algorithms equal to $1.63 \cdot 10^{-12}$, and the maximum norm of the difference between eigenvectors multiplied by the corresponding relative gap has been $1.04 \cdot 10^{-12}$. Both of them occurred for $n = 2000$. In a second type of RRDs, we repeat the same experiment with the only modification that the option `MODE = 1` in `gallery('randsvd',...)` is used to generate the absolute values of the diagonal factors D. The average numbers of sweeps for these RRDs are presented in Table 4. We have observed in the matrices in Table 4 a maximum relative difference between the eigenvalues computed by Algorithm 1 and those computed by the other algorithms equal to $6.39 \cdot 10^{-13}$, and the maximum norm of the difference between eigenvectors multiplied by the corresponding relative gap has been $5.83 \cdot 10^{-13}$. Both of them occurred again for $n = 2000$.

One can obtain from Test 4 similar conclusions to those obtained from Test 3 on the comparison of the different algorithms.

Table 4 Average numbers of Jacobi sweeps for random $n \times n$ RRDs with the singular values of D generated with `MODE=1`, $\kappa(D) = 10^{40}$, and $\kappa(X) = 100$.

n	Algor. 1	Algor. 3	J-orth	SSVD
100	10.8	8.8	11.6	7.8
500	12.8	12	13.2	9.6
1000	13	13	14	10.7
2000	14.5	13.5	15	11

10 Conclusions

We have introduced the first algorithm that computes with guaranteed high relative accuracy the eigenvalues and eigenvectors of any symmetric indefinite (or definite) matrix in RRD form, $A = XDX^T$, by using only orthogonal transformations and respecting the symmetry of the problem. This algorithm simply applies the rotations of the standard cyclic-by-row Jacobi algorithm implicitly on the factor X. A rigorous error analysis proving the high relative accuracy obtained by this algorithm has been developed. This error analysis is based on new theoretical results on properties of diagonal and scaled diagonally dominant symmetric RRDs that show that disastrous cancellations do not appear in the computation of the eigenvalues. Numerical tests have been performed to confirm the high relative accuracy of the new implicit Jacobi algorithm. The new algorithm can be easily preconditioned through the QR decomposition with column pivoting. This preconditioned version preserves all the good properties of the implicit Jacobi algorithm and runs much faster, as it has been shown in Section 9. The computational cost of the preconditioned implicit Jacobi algorithm is similar to the nonsymmetric SSVD algorithm, which is at present the fastest existing algorithm for computing eigenvalues and eigenvectors of symmetric indefinite RRDs with guaranteed high relative accuracy. In future work we will consider how to speed up the new algorithm preserving its three fundamental properties: guaranteed error bounds, preservation of the symmetry, and using only orthogonal transformations. This may require much more sophisticated ideas in the spirit of the ones presented in [20,21,26,27] for the accurate computation of the Singular Value Decomposition.

Acknowledgements The authors thank J. Demmel, Z. Drmač, and J. Moro for many useful discussions on accurate eigensolvers for symmetric indefinite problems. The authors also thank the Editor of this manuscript, I. Ipsen, and two referees for many useful suggestions that have helped us to improve the presentation, as well as, to improve some of the results in Section 4. Moreover, we want to remark that one of these referees, Z. Drmač, suggested the preconditioning technique included in Section 8. This technique plays an essential role to speed-up the implicit Jacobi algorithm and to make it useful in practice.

References

1. Anda, A., Park, H.: Fast plane rotations with dynamic scaling. SIAM J. Matrix Anal. Appl. **15**, 162–174 (1994)
2. Anderson, E., Bai, Z., Bischof, C., Blackford, S., Demmel, J., Dongarra, J., Du Croz, J., Greenbaum, A., Hammarling, S., McKenney, A., Sorensen, D.: LAPACK Users' Guide, Third Edition, Software Environ. Tools 9. SIAM, Philadelphia (1999)
3. Barlow, J., Demmel, J.: Computing accurate eigensystems of scaled diagonally dominant matrices. SIAM J. Num. Anal. **27**(3), 762–791 (1990)
4. Businger, P., Golub, G.H.: Linear least squares solutions by Householder transformations. Numer. Math. **7**, 269–276 (1965)

5. Demmel, J.: Accurate singular value decompositions of structured matrices. SIAM J. Matrix Anal. Appl. **21**(2), 562–580 (1999)
6. Demmel, J., Gragg, W.: On computing accurate singular values and eigenvalues of acyclic matrices. Linear Algebra Appl. **185**, 203–218 (1993)
7. Demmel, J., Gu, M., Eisenstat, S., Slapničar, I., Veselić, K., Drmač, Z.: Computing the singular value decomposition with high relative accuracy. Linear Algebra Appl. **299**(1–3), 21–80 (1999)
8. Demmel, J., Kahan, W.: Accurate singular values of bidiagonal matrices. SIAM J. Sci. Statist. Comput. **11**(5), 873–912 (1990)
9. Demmel, J., Koev, P.: Necessary and sufficient conditions for accurate and efficient rational function evaluation and factorizations of rational matrices. In: Structured matrices in mathematics, computer science, and engineering, II (Boulder, CO, 1999), *Contemp. Math.*, vol. 281, pp. 117–143. Amer. Math. Soc., Providence, RI (2001)
10. Demmel, J., Koev, P.: Accurate SVDs of weakly diagonally dominant M-matrices. Numer. Math. **98**(1), 99–104 (2004)
11. Demmel, J., Koev, P.: Accurate SVDs of polynomial Vandermonde matrices involving orthonormal polynomials. Linear Algebra Appl. **417**(2-3), 382–396 (2006)
12. Demmel, J., Veselić, K.: Jacobi's method is more accurate than QR. SIAM J. Matrix Anal. Appl. **13**(4), 1204–1246 (1992)
13. Demmel, J.W.: Applied Numerical Linear Algebra. SIAM, Philadelphia (1997)
14. Dopico, F.M., Koev, P.: Accurate symmetric rank revealing and eigendecompositions of symmetric structured matrices. SIAM J. Matrix Anal. Appl. **28**(4), 1126–1156 (2006)
15. Dopico, F.M., Molera, J.M.: Perturbation theory for factorizations of LU type through series expansions. SIAM J. Matrix Anal. Appl. **27**(2), 561–581 (2005)
16. Dopico, F.M., Molera, J.M., Moro, J.: An orthogonal high relative accuracy algorithm for the symmetric eigenproblem. SIAM J. Matrix Anal. Appl. **25**(2), 301–351 (2003)
17. Drmač, Z.: Implementation of Jacobi rotations for accurate singular value computation in floating point arithmetic. SIAM J. Sci. Comput. **18**(4), 1200–1222 (1997)
18. Drmač, Z.: Accurate computation of the product induced singular value decomposition with applications. SIAM J. Num. Anal. **35**(5), 1969–1994 (1998)
19. Drmač, Z.: A posteriori computation of the singular vectors in a preconditioned Jacobi SVD algorithm. IMA J. Num. Anal. **19**, 191–213 (1999)
20. Drmač, Z., Veselić, K.: New fast and accurate Jacobi SVD algorithm. I. SIAM Journal on Matrix Analysis and Applications **29**(4), 1322–1342 (2008)
21. Drmač, Z., Veselić, K.: New fast and accurate Jacobi SVD algorithm. II. SIAM Journal on Matrix Analysis and Applications **29**(4), 1343–1362 (2008)
22. Eisenstat, S., Ipsen, I.: Relative perturbation techniques for singular value problems. SIAM J. Numer. Anal. **32**(6), 1972–1988 (1995)
23. Fernando, K., Parlett, B.: Accurate singular values and differential qd algorithms. Numer. Math. **67**, 191–229 (1994)
24. Gentleman, W.M.: Error analysis of QR decompositions by Givens transformations. Linear Algebra and Appl. **10**, 189–197 (1975)
25. Golub, G., Van Loan, C.: Matrix Computations, 3rd edn. Johns Hopkins University Press, Baltimore, MD (1996)
26. Hari, V.: Accelerating the SVD block-Jacobi method. Computing **75**(1), 27–53 (2005)
27. Hari, V.: Convergence of a block-oriented quasi-cyclic Jacobi method. SIAM J. Matrix Anal. Appl. **29**(2), 349–369 (2007)
28. Higham, N.J.: The Matrix Computation Toolbox. `http://www.ma.man.ac.uk/~higham/mctoolbox`
29. Higham, N.J.: Accuracy and Stability of Numerical Algorithms, Second edn. SIAM, Philadelphia (2002)
30. Horn, R.A., Johnson, C.R.: Topics in matrix analysis. Cambridge University Press, Cambridge (1994). Corrected reprint of the 1991 original
31. Koev, P.: Accurate eigenvalues and SVDs of totally nonnegative matrices. SIAM J. Matrix Anal. Appl. **27**(1), 1–23 (2005)
32. Koev, P., Dopico, F.: Accurate eigenvalues of certain sign regular matrices. Linear Algebra Appl. **424**(2-3), 435–447 (2007)
33. Li, R.C.: Relative perturbation theory. II. Eigenspace and singular subspace variations. SIAM J. Matrix Anal. Appl. **20**(2), 471–492 (1999)
34. Li, R.C.: Relative perturbation theory. IV. $\sin 2\theta$ theorems. Linear Algebra Appl. **311**(1-3), 45–60 (2000)
35. Mathias, R.: Accurate eigensystem computations by Jacobi methods. SIAM J. Mat. Anal. Appl. **16**(3), 977–1003 (1995)
36. The MathWorks, Inc., Natick, MA: MATLAB Reference Guide (1992)
37. Parlett, B.N.: The symmetric eigenvalue problem. SIAM, Philadelphia (1998)

38. Peláez, M.J., Moro, J.: Accurate factorization and eigenvalue algorithms for symmetric DSTU and TSC matrices. SIAM J. Matrix Anal. Appl. **28**(4), 1173–1198 (2006)
39. Rutishauser, H.: The Jacobi method for real symmetric matrices. Numer. Math. **9**(1), 1–10 (1966)
40. Slapničar, I.: Componentwise analysis of direct factorization of real symmetric and Hermitian matrices. Linear Algebra Appl. **272**, 227–275 (1998)
41. Slapničar, I.: Highly accurate symmetric eigenvalue decomposition and hyperbolic SVD. Linear Algebra Appl. **358**, 387–424 (2003). Special issue on accurate solution of eigenvalue problems (Hagen, 2000)
42. Slapničar, I.: Accurate symmetric eigenreduction by a Jacobi method. Ph.D. thesis, Fernuniversität - Hagen, Hagen, Germany (1992)
43. Stewart, G.W., Sun, J.G.: Matrix Perturbation Theory. Academic Press, New York (1990)
44. Veselić, K.: A Jacobi eigenreduction algorithm for definite matrix pairs. Num. Math. **64**, 241–269 (1993)
45. Ye, Q.: Computing singular values of diagonally dominant matrices to high relative accuracy. Math. Comp. **77**(264), 2195–2230 (2008)