SIAM J. MATRIX ANAL. APPL.
Vol. 25, No. 4, pp. 1021–1031

A NOTE ON MULTIPLICATIVE BACKWARD ERRORS OF ACCURATE SVD ALGORITHMS*

FROILÁN M. DOPICO† AND JULIO MORO†

Abstract. Multiplicative backward stability results are presented for two algorithms which compute the singular value decomposition of dense matrices. These algorithms are the classical one-sided Jacobi algorithm, with a stringent stopping criterion, and an algorithm which uses one-sided Jacobi to compute high accurate singular value decompositions of matrices given as rank-revealing factorizations. When multiplicative backward errors are small, the multiplicative perturbation theory for the singular value decomposition developed in the last decade can be applied to get high accuracy bounds on the errors of the computed singular values and vectors.

Key words. singular value decomposition, Jacobi algorithm, high relative accuracy, rank-revealing decompositions, mutiplicative perturbation theory

AMS subject classifications. 65F15, 65G50

DOI. 10.1137/S0895479803427005

1. Introduction.

The singular value decomposition (SVD) of a matrix $G \in \mathbb{R}^{m\times n}$ $(m \geq n)$ is the factorization $G = U\Sigma V^T$, where $U \in \mathbb{R}^{m\times n}$ has orthonormal columns, $V \in \mathbb{R}^{n\times n}$ is an orthogonal matrix, and $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_n)$ is nonnegative and diagonal. The columns of U are the left singular vectors of G, the columns of V are the right singular vectors of G, and σ_i are the singular values of G. Given two nonsingular square matrices D_1 and D_2, the matrix D_1GD_2 is called a multiplicative perturbation of G. In the last decade, a perturbation theory bounding the differences between the singular values and vectors of G and D_1GD_2 has been developed [13, 18, 19, 17]. Let $\sigma_1 \geq \cdots \geq \sigma_n \geq 0$ and $\tilde{\sigma}_1 \geq \cdots \geq \tilde{\sigma}_n \geq 0$ be, respectively, the singular values of G and D_1GD_2 and u_i, v_i, $\tilde{u}_i, \tilde{v}_i$, $i = 1, \ldots, n$, be the corresponding pairs of left and right singular vectors. Let us denote by $\|\,.\,\|$ the usual Euclidean vector norm when the argument is a vector and the spectral, or two, matrix norm when the argument is a matrix. Then the multiplicative perturbation theory essentially bounds

$$\frac{|\sigma_i - \tilde{\sigma}_i|}{\sigma_i} \quad \text{and} \quad \max\{\|v_i - \tilde{v}_i\|, \|u_i - \tilde{u}_i\|\}\mathrm{relgap}_i, \qquad i = 1, \ldots, n, \tag{1}$$

where $\mathrm{relgap}_i = \min_{j\neq i} |\sigma_i - \tilde{\sigma}_j|/\sigma_i$, by a small integer constant times $\max\{\|I - D_1\|, \|I - D_2\|\}$ [13, 19]. Therefore, if D_1 and D_2 are close to the identity matrix, the relative differences between the singular values of G and D_1GD_2 are small, and the differences between the singular vectors multiplied by the relative gaps are also small. Obviously, (1) makes sense only if $\sigma_i \neq 0$. If $\sigma_i = 0$, then it is trivial that $\tilde{\sigma}_i = 0$ and it can be shown that the differences between the corresponding singular vectors are simply less than a small integer constant times $\max\{\|I - D_1\|, \|I - D_2\|\}$ [13, 19]. Notice that classical perturbation theory [22], valid for additive perturbations of the type $G + E$, bounds absolute differences between singular values, i.e., $|\sigma_i - \tilde{\sigma}_i| \leq \|E\|$,

*Received by the editors April 30, 2003; accepted for publication (in revised form) by I.C.F. Ipsen October 9, 2003; published electronically June 4, 2004. The research conducted for this paper was partially supported by the Ministerio de Ciencia y Tecnología of Spain through grant BFM-2000-0008.

http://www.siam.org/journals/simax/25-4/42700.html

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

and the gaps appearing in the singular vector bounds are also absolute, i.e., $\text{gap}_i = \min_{j \neq i} |\sigma_i - \tilde{\sigma}_j|/\sigma_1$.

Multiplicative perturbation theory has been successfully used in proving that some algorithms are able to compute the SVD with high relative accuracy when applied to matrices with special structure. Here high relative accuracy means that the relative errors in the computed singular values are of order ϵ, with ϵ being the machine precision, and that the errors in the computed singular vectors are of order ϵ divided by the corresponding singular value relative gap, i.e., relgap_i. Well-known examples of matrices for which it is possible to compute high relative accuracy SVDs are bidiagonal matrices [5, 14]; matrices of the form $G = BD$, with D diagonal and B well-conditioned [6, 11, 20]; positive definite matrices of the form DAD, with D diagonal and A well-conditioned [6]; and matrices for which it is possible to compute accurately a rank-revealing decomposition [4]. This latter class contains the previous ones and many others (see also [3, 7, 8]). A technical remark is in order here: although the approach in [4] includes the case of bidiagonal matrices, since bidiagonal matrices are acyclic, the original approaches in [5, 14] are much faster and do not require one to compute a rank-revealing factorization.

There exists a relative perturbation theory for additive perturbations which gives structured bounds for the quantities appearing in (1); see [17] and references therein. This perturbation theory has been used to guarantee the high relative accuracy of the SVDs computed by some algorithms [6, 11, 20]. However, it was shown in [4] that multiplicative perturbation theory can also be used in these cases. Thus, at present, it seems that multiplicative perturbation theory has a wider applicability in the context of high relative accuracy computations of SVDs. In fact, multiplicative perturbation theory and some of its applications have already been presented in some recent text books [2, sections 5.2.1, 5.4.2, 5.4.3].

Although the accurate computation of the SVD is still a work in progress and, as a consequence, it is still too early to know which tools will be the most useful in future developments, there are sound reasons to support the prominent role of multiplicative perturbation theory: for instance, the simplicity of the bounds, or the simple way in which multiplicative perturbation bounds can be composed with each other.

In spite of the present importance of multiplicative perturbation theory, there is no theorem so far stating in multiplicative form the backward stability properties of high accuracy algorithms for the SVD, i.e., a theorem saying that the computed SVD of a matrix G is essentially the exact SVD of a nearby *multiplicative* perturbation of G. In the context of usual algorithms for SVD computations the usual backward stability result [1, section 4.9.1] states that the computed SVD of a matrix G is essentially the exact SVD of a nearby *additive* perturbation of G, i.e., a matrix $G + E$ with $\|E\| \leq p(m,n)\,\epsilon\,\|G\|$ and $p(m,n)$ a modestly growing function of m and n. Our goal in this note is to prove a very strong form of multiplicative backward stability for two algorithms which are able to compute SVDs with high relative accuracy in some important cases. The starting point will be the roundoff error analysis previously developed by other authors in [4, 6], and especially in [11]. The theorems we obtain have already been used in [9] and greatly simplify the way in which the error bounds for singular values and vectors are obtained in [6, 20, 4] just by using the multiplicative perturbation theory for the SVD. Moreover, we hope that the theorems we present will be useful in future error analyses of accurate SVD algorithms.

Finally, it is interesting to stress that the multiplicative backward error results we are going to present cannot be deduced from additive backward error results of

the form $G + E$ just by factoring out the inverse or pseudoinverse of G. This is obvious in the case of standard backward stability results [1, section 4.9.1], because the information about the perturbation is just $\|E\| \leq p(m,n)\,\epsilon\,\|G\|$. Therefore, if we write $G + E = G(I + G^{-1}E)$, the most we can assert on the magnitude of the multiplicative perturbation is $\|G^{-1}E\| \leq p(m,n)\,\epsilon\,\|G^{-1}\|\,\|G\|$, and the condition number $\|G^{-1}\|\,\|G\|$ can be very large. On the other hand, factoring out G in the additive backward error result appearing in [11, Proposition 3.13] for the one-sided Jacobi algorithm will play an essential role in our developments, but this is not the only thing to do. In fact, we will need to introduce multiplicative perturbations on *both* sides of the matrix. This is the reason why the stability results presented in [6, 11] are mixed forward-backward error results.

The paper is organized as follows: a multiplicative backward stability theorem is proved in section 2 for the one-sided Jacobi algorithm, and the same is done for Algorithm 3.1 of [4] in section 3. Finally, in section 4 we discuss a different version of one-sided Jacobi, which is usually faster although the error bounds are weaker.

Notation and model of arithmetic. In the statements of the subsequent theorems big-O notation will be used. Given a scalar quantity b, the meaning of $O(\epsilon b)$ is that $O(\epsilon b) = p(m,n)\,\epsilon\,b + O(\epsilon^2)$ with $p(m,n)$ a polynomial of low degree in the dimensions m, n of the problem.

The conventional error model for floating point arithmetic with guard-digit will be used:

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

where a and b are real floating point numbers, $\odot \in \{+, -, \times, /\}$, and $|\delta| \leq \epsilon$, where ϵ is the machine precision. Moreover, we assume that neither overflow nor underflow occur. For the sake of simplicity, we will commit a slight abuse of notation, denoting by $\texttt{fl}(expr)$ the computed result in finite precision of expression $expr$, instead of its rigorous meaning of the closest floating point number to $expr$.

2. Backward error of one-sided Jacobi SVD algorithm. One-sided Jacobi algorithms for the SVD [15, section 8.6.3] multiply a matrix by a sequence of Jacobi rotations, all of them acting on the same side. When the rotations are applied to the matrix from the left (right), the goal is to converge to a matrix with orthogonal rows (columns). These two different implementations of one-sided Jacobi will be called, respectively, left-handed and right-handed Jacobi. A detailed pseudocode for the right-handed Jacobi algorithm can be found in [6, Algorithm 4.1]. The left-handed version follows easily from the right-handed version applied to the transpose matrix.

A plain implementation of one-sided Jacobi yields an algorithm much slower than the SVD algorithms based on first bidiagonalizing the matrix. However, one-sided Jacobi has an important advantage: if the stopping criterion proposed in [6, Algorithm 4.1] is used, then the one-sided Jacobi algorithm is able to compute the SVD with high relative accuracy for matrices that are the product of a diagonal matrix (possibly with elements of widely varying magnitudes) and a well-conditioned matrix. To be more precise, let D be a diagonal matrix; then high relative accuracy is achieved for matrices of the type DB if B has full row rank and is well-conditioned, or BD if B has full column rank and is well-conditioned. This high relative accuracy was first proved in [6] under a minor proviso. A proof valid in general was presented in [11] (see also references therein) and [20]. In this latter proof it is essential that the Jacobi rotations are applied on the side opposite to the diagonal matrix D. At present,

fast and sophisticated versions of one-sided Jacobi algorithm are being developed by Drmač along the ideas of [12].

It is very important to remark that if one-sided Jacobi is implemented as in [6, Algorithm 4.1], then underflows appear frequently for very ill conditioned matrices, and the high relative accuracy in the computed SVD expected for matrices of the form DB or BD (see previous paragraph) is lost. To get results with high relative accuracy, whenever the singular values are inside the range of the arithmetic, the Jacobi rotations have to be carefully implemented according to the method developed in [10].

The next theorem proves that the one-sided Jacobi SVD algorithm on a square invertible matrix produces a small multiplicative backward error; i.e., the computed SVD is nearly the exact SVD of a close multiplicative perturbation of the original matrix. We restrict ourselves to square matrices because, in practice, for the nonsquare case a QR factorization is computed first, and then one-sided Jacobi is applied to the square factor R. This reduces the computational cost. The following notation will be used: the ith column (resp., row) of any matrix A is denoted by $A(:,i)$ (resp., $A(i,:)$), $\widetilde{A}$ denotes the last matrix in the sequence computed by the right-handed Jacobi process, and $\kappa(A)$ is the spectral condition number of A. This theorem is based on the error analysis presented in [11, Proposition 3.13] and shows that with a small additional effort a strong backward multiplicative result can be obtained.

THEOREM 2.1. *Let $A \in \mathbb{R}^{n\times n}$ be an invertible matrix and let $\widehat{U}\widehat{\Sigma}\widehat{V}^T$ be the SVD computed in finite arithmetic with machine precision ϵ by the right-handed Jacobi SVD algorithm applied on A with stopping criterion*[1]

$$\max_{i\neq j} \mathtt{fl}\left(\frac{|\widetilde{A}(:,i)^T\widetilde{A}(:,j)|}{\|\widetilde{A}(:,i)\|\,\|\widetilde{A}(:,j)\|}\right) \leq n\,\epsilon \qquad \text{for} \quad i\neq j. \tag{2}$$

Then there exist matrices U', V', E_L, $E_R \in \mathbb{R}^{n\times n}$, such that U' and V' are orthogonal,

$$\begin{aligned} &\|U'-\widehat{U}\| = O(\epsilon), \quad \|V'-\widehat{V}\| = O(\epsilon),\\ &\|E_L\| = O(\epsilon), \quad \|E_R\| = O(\epsilon\kappa(A_N)), \end{aligned} \tag{3}$$

where $A_N = D_N^{-1}A$, with D_N a diagonal matrix with elements $(D_N)_{ii} = \|A(i,:)\|$, and

$$(I+E_L)A(I+E_R) = U'\widehat{\Sigma}V'^T. \tag{4}$$

Proof. It is known [11, Proposition 3.13] that, under the conditions above, the matrix $\widetilde{A}$ satisfying the stopping criterion (2) can be written as

$$\widetilde{A} = (A+\delta A)V'$$

for an orthogonal matrix V' with $\|V'-\widehat{V}\| = O(\epsilon)$ and δA such that

$$\|\delta A(i,:)\| \leq \epsilon_J\,\|A(i,:)\|, \qquad i=1,\ldots,n, \tag{5}$$

[1] A similar result holds with $n\epsilon$ replaced by any tolerance *tol* in criterion (2). In that case, $\|U'-\widehat{U}\| \leq n\,tol + O(\epsilon)$ and $\|E_L\| \leq n\,tol + O(\epsilon)$. Notice, however, that if the tolerance is larger than $O(\epsilon)$, then the computed left singular vectors will fail, in general, to be orthogonal up to $O(\epsilon)$.

for a certain $\epsilon_J = O(\epsilon)$ which depends on the sweeps required for convergence.[2] Hence,

$$\widetilde{A} = A(I + E_R)V' \tag{6}$$

for $E_R = A^{-1}\delta A$. If we now scale $A = D_N A_N$, so that A_N has rows of unit Euclidean length, the bound (5) implies

$$\|E_R\|_F \leq \|A_N^{-1}\|_F \, \|D_N^{-1}\delta A\|_F \leq \sqrt{n}\,\epsilon_J \, \|A_N^{-1}\|_F,$$

where $\|\cdot\|_F$ stands for the Frobenius norm.[3] Finally, since $\|A_N\|_F = \sqrt{n}$, it follows that the Frobenius norm of E_R, and consequently its spectral norm, is bounded by $\epsilon_J\,\kappa_F(A_N) = O(\epsilon\kappa(A_N))$.

On the other hand, recall that if we denote by $\widetilde{\Sigma}$ the diagonal matrix whose ith diagonal entry is the Euclidean norm of the ith column of $\widetilde{A}$, then $\widehat{\Sigma}$ and $\widehat{U}$ are computed as $\widehat{\Sigma} = \mathtt{fl}(\widetilde{\Sigma})$ and $\widehat{U} = \mathtt{fl}(\widetilde{A}\widehat{\Sigma}^{-1})$. Notice that each element $\widehat{u}_{ij}$ of $\widehat{U}$ can be written as $\widehat{u}_{ij} = (\widetilde{A}_{ij}/\widehat{\Sigma}_{jj})(1+\epsilon_{ij})$ with $|\epsilon_{ij}| \leq \epsilon$. Let U be the matrix such that $\widetilde{A} = U\,\widehat{\Sigma}$. Then (6) implies that

$$U\widehat{\Sigma}(V')^T = A(I + E_R)$$

with $\|U - \widehat{U}\|_F \leq \epsilon\|U\|_F$. It remains only to show, using the stopping criterion, that there is an orthogonal matrix U' such that

$$U = (I + E_L)^{-1}U'$$

with $\|E_L\| = O(\epsilon)$ and $\|U' - \widehat{U}\| = O(\epsilon)$.

It follows from condition (2) that each off-diagonal element of U^TU is bounded in absolute value by $cn\epsilon + O(\epsilon^2)$, with c a small integer constant. The diagonal elements of U^TU, on the other hand, are $1 + \alpha_{ii}$ with $|\alpha_{ii}| \leq cn\epsilon + O(\epsilon^2)$. Thus, $\|U^TU - I\|_F \leq cn^2\epsilon + O(\epsilon^2)$. If $U = W_L(I + \delta\Sigma)W_R^T$ is the SVD of U, then $\|\delta\Sigma\|_F \leq cn^2\epsilon + O(\epsilon^2)$. Denoting $U' = W_LW_R^T$, it follows that $U = (I + \delta U)U'$, where U' is orthogonal and $\|\delta U\|_F = \|\delta\Sigma\|_F$.

Defining $E_L = (I + \delta U)^{-1} - I$, we obtain that $\|E_L\|_F = \|\delta U\|_F + O(\|\delta U\|_F^2) \leq cn^2\epsilon + O(\epsilon^2)$.

Finally, $\|\widehat{U} - U'\|_F \leq \|\widehat{U} - U\|_F + \|U - U'\|_F$, but $\|U - U'\|_F = \|\delta U\|_F \leq cn^2\epsilon + O(\epsilon^2)$, and $\|\widehat{U} - U\|_F \leq \epsilon\|U\|_F \leq \sqrt{n}\epsilon + O(\epsilon^2)$. ∎

As explained in the introduction, applying multiplicative perturbation results to (4) yields relative error bounds on the singular values of order $O(\epsilon\kappa(A_N))$ and of order $O(\epsilon\kappa(A_N))$ divided by the relative gaps in the singular vectors. Thus, the magnitude of $\kappa(A_N)$ gives the relative accuracy of the computed SVD. In this respect, recall that $\kappa(A_N) \leq \sqrt{n}\min\kappa(D\,A)$, with D any diagonal matrix [21].

3. Backward error of a SVD algorithm for rank-revealing decompositions. A *rank-revealing decomposition* (RRD) [4] of $G \in \mathbb{R}^{m\times n}$, $m \geq n$, is a factorization $G = XDY^T$ with $D \in \mathbb{R}^{r\times r}$ diagonal and nonsingular and $X \in \mathbb{R}^{m\times r}$, $Y \in$

[2] Admittedly, it is not fully true that $\epsilon_J = O(\epsilon)$ with the meaning we have given to $O(\epsilon)$ in the *Notation*, since a dependence in the number of steps required for the convergence of the algorithm is hidden in the constant of the $O(\epsilon)$ (see [11, Proposition 3.13]). However, extensive numerical experience indicates that this dependence is polynomic in the dimensions of the problem.

[3] One can also show that $\|E_R\| \leq \sqrt{n}\,\epsilon_J\,\|A_N^{-1}\|$ in the spectral norm.

$\mathbb{R}^{n\times r}$, where both matrices X, Y have full column rank and are well-conditioned (notice that this implies $r = \mathrm{rank}(G)$). One of the most important contributions of Demmel et al. in [4] is developing algorithms which compute high relative accuracy SVDs for any matrix such that an RRD can be computed with enough accuracy. The accuracy required in the computed $\widehat{X}, \widehat{D}$ and $\widehat{Y}$ is the following (see [4, Theorem 2.1]):

1. each entry of D has small relative error,

$$|D_{ii} - \widehat{D}_{ii}| \leq O(\epsilon)|D_{ii}|, \tag{7}$$

2. $\widehat{X}$ and $\widehat{Y}$ have small norm errors,

$$\|X - \widehat{X}\| = O(\epsilon)\|X\| \quad \text{and} \quad \|Y - \widehat{Y}\| = O(\epsilon)\|Y\|. \tag{8}$$

Once the RRD is computed, Algorithms 3.1 or 3.2 in [4] can be used to compute the SVD with high relative accuracy. Both algorithms have as inputs the three factors, X, D and Y, of an RRD. The error bounds for the computed SVD presented in [4] for Algorithm 3.2 are better than those proved for Algorithm 3.1. However, the authors of [4] strongly recommend the use of Algorithm 3.1. The reasons are that Algorithm 3.1 is faster and that no significant difference in accuracy is observed in practice.

In this section we prove that Algorithm 3.1 in [4] produces a small backward multiplicative error when executed in finite precision arithmetic. This result is based on the proof of Theorem 3.1 in [4, section 3.2.1] and greatly clarifies the way in which the error bounds for the computed singular values and vectors are obtained in [4]. The error analysis done in [4] is backward multiplicative up to the one-sided Jacobi step of Algorithm 3.1 in [4]. From this point on the analysis is made in the forward sense and becomes quite involved. The crucial ingredient to get a multiplicative backward error result like Theorem 3.1 below is Theorem 2.1 for one-sided Jacobi proved in section 2.

Algorithm 1 below is the version of Algorithm 3.1 in [4] we analyze. We stress that the inputs for Algorithm 1 are the three matrices $X \in \mathbb{R}^{m\times r}$, $D \in \mathbb{R}^{r\times r}$, $Y \in \mathbb{R}^{n\times r}$ of a RRD. Moreover, the QR and LQ factorizations appearing in the Algorithm are economy size or reduced factorizations, i.e., if $C = QR$ is a $n \times r$ matrix ($n > r$), then Q is a $n \times r$ matrix with orthonormal columns.

Algorithm 1.

Input: rank-revealing decomposition, X, D, Y, of $G = XDY^T \in \mathbb{R}^{m\times n}$.
Output: singular value decomposition $U\Sigma V^T$ of G.

1. *Compute a QR decomposition with column pivoting, $XD = QRP$, of XD.*
2. *Compute the product $W = RPY^T$ using conventional matrix multiplication.*
3. *Compute a LQ decomposition $W = L_\omega Q_\omega^T$ of W.*
4. *Compute an SVD $L_\omega = U_\omega \Sigma V_\omega^T$ of L_ω using right-handed Jacobi.*
5. *Compute the products $U = QU_\omega$ and $V = Q_\omega V_\omega$. Strassen's method may be used.*

We should point out that this implementation differs from the one presented in [4]: here the Jacobi step is split in two stages, steps 3 and 4. This is recommended in [4, section 3.3] to reduce the computational cost of the one-sided Jacobi step, the most expensive one in the whole algorithm. This saving is clear if the rank r is less than n. In the case $r = n$, W is square and, at first glance, the computation of the

LQ factorization of W would increase the cost because right-handed Jacobi does not make any use of the triangular form of L_ω. However, if the LQ factorization of W is done with row pivoting, then numerical experience shows that more than one sweep is saved in right-handed Jacobi. This is enough to compensate the cost of the LQ factorization and makes step 3 of Algorithm 1 still interesting. Anyway, the reader can check that skipping step 3 above does not affect the error bounds in Theorem 3.1.

THEOREM 3.1. *Algorithm* 1 *produces a small multiplicative backward error; i.e., if* $\widehat{U}\widehat{\Sigma}\widehat{V}^T$ *is the SVD computed by the algorithm in finite arithmetic with machine precision* ϵ*, then there exist matrices* $U' \in \mathbb{R}^{m\times r}$, $V' \in \mathbb{R}^{n\times r}$, $E \in \mathbb{R}^{m\times m}$, $F \in \mathbb{R}^{n\times n}$ *such that* U' *and* V' *have orthonormal columns,*

$$\begin{aligned} \|U' - \widehat{U}\| = O(\epsilon), \quad \|V' - \widehat{V}\| = O(\epsilon), \\ \|E\| = O(\epsilon\kappa(X)), \quad \|F\| = O(\epsilon\kappa(R')\kappa(Y)), \end{aligned} \tag{9}$$

where R' *is the best conditioned row diagonal scaling of the triangular matrix* R *appearing in step* 1 *of Algorithm* 1 *and*

$$(I+E)G(I+F) = U'\widehat{\Sigma}V'^T. \tag{10}$$

Remark 1. It is proved in [4] that $\kappa(R')$ is at most of order $O(n^{3/2}\kappa(X))$, but in practice extensive numerical tests show that $\kappa(R')$ behaves as $O(n)$ [4, 9]. One can get rid of the factor $\kappa(R')$ at the price of using the more costly Algorithm 3.2 of [4]. The proof of this follows closely the proof of Theorem 3.1.

Proof. Since we will use results in [4, section 3.2.1], we need to match our notation with that of [4]: the matrices Q, W, R (and R') appearing in the proof, which are the computed ones, are named in the proof *without* hats. The rest of the computed matrices are denoted, as elsewhere in this paper, with their hats on.

It is shown in [4, p. 34] that, after step 2 of Algorithm 1, the matrix Q computed in step 1 and the matrix W computed in step 2 are such that

$$(I+E_1)G(I+F_1) = QW \tag{11}$$

for square matrices E_1, F_1 with

$$\|E_1\| = O(\epsilon\,\kappa(X)), \qquad \|F_1\| = O(\epsilon\,\kappa(R')\,\kappa(Y)).$$

Although the columns of the computed Q are not exactly orthonormal, it is well known [16, p. 360] that there exists a matrix Q' with orthonormal columns such that

$$Q = Q' + E_q = (I + E_q(Q')^T)Q', \tag{12}$$

with $\|E_q\| = O(\epsilon)$. Thus, (11) becomes $(I+E_1')G(I+F_1) = Q'W$, with $\|E_1'\| = O(\epsilon\,\kappa(X))$.

The LQ factorization of W in step 3 of Algorithm 1 is equivalent to computing a QR factorization of $W^T \in \mathbb{R}^{n\times r}$. The usual additive backward error analysis of the QR factorization, applied columnwise [16, p. 360], ensures that the computed $\widehat{L}_\omega$ satisfies

$$\widehat{L}_\omega(Q'_\omega)^T = (W + E_\omega),$$

where $Q'_\omega \in \mathbb{R}^{n\times r}$ is a matrix with orthonormal columns satisfying $\|Q'_\omega - \widehat{Q}_\omega\| = O(\epsilon)$ for the computed $\widehat{Q}_\omega$. The backward error E_ω satisfies the rowwise bound

$$\|E_\omega(i,:)\| = O(\varepsilon)\|W(i,:)\|, \qquad i = 1,\ldots,r. \tag{13}$$

If we write $W + E_\omega = W(I + W^\dagger E_\omega)$ multiplicatively, with $W^\dagger$ the pseudoinverse of W, then

$$W = \widehat{L}_\omega (Q'_\omega)^T (I + W^\dagger E_\omega)^{-1}.$$

Now, let $R' = (D')^{-1}R$ be the best conditioned row scaling of the triangular matrix R computed in step 1. In order to bound $\|W^\dagger E_\omega\|$, we define $Z = (D')^{-1}W$ and $E_z = (D')^{-1}E_\omega$. The equations (13) imply $\|E_z\| = O(\epsilon)\|Z\|$, and since both D' and Z have full rank, we obtain

$$\|W^\dagger E_\omega\| = \|Z^\dagger E_z\| = O(\epsilon)\kappa(Z) = O(\epsilon\kappa(R')\kappa(Y)).$$

The last equality above is a consequence of the first equation in [4, p. 34], which implies $\|(D')^{-1}\delta W\| = O(\epsilon)\|R'\|\,\|Y\|$ for the error δW in the matrix multiplication of step 2 of Algorithm 1. Therefore, since $Z = R'PY^T - (D')^{-1}\delta W$, we arrive at $\kappa(Z) \leq \kappa(R')\kappa(Y)(1 + O(\epsilon)\kappa(R')\kappa(Y))$.

Thus, upon completion of step 3 of Algorithm 1, we have

$$(I + E_2)G(I + F_2) = Q'\,\widehat{L}_\omega (Q'_\omega)^T \tag{14}$$

with $E_2 = E'_1$, $I + F_2 = (I + F_1)(I + W^\dagger E_\omega)$ and $\|F_2\| = O(\epsilon\kappa(R')\kappa(Y))$.

Now, Theorem 2.1 applied to step 4 ensures the existence of $r \times r$ matrices $\overline{U}'$, $\overline{V}'$, E_L, E_R with $\overline{U}'$, $\overline{V}'$ orthogonal,

$$\begin{aligned} &\|\overline{U}' - \widehat{U}_\omega\| = O(\epsilon), \qquad \|\overline{V}' - \widehat{V}_\omega\| = O(\epsilon) \\ &\|E_L\| \leq O(\epsilon), \qquad \|E_R\| \leq O(\epsilon\kappa((D')^{-1}\widehat{L}_\omega)), \end{aligned} \tag{15}$$

and

$$\widehat{L}_\omega = (I + E_L)\overline{U}'\widehat{\Sigma}(\overline{V}')^T(I + E_R), \tag{16}$$

where $\widehat{U}_\omega\widehat{\Sigma}\widehat{V}_\omega^T$ is the SVD computed by the right-handed Jacobi SVD algorithm on $\widehat{L}_\omega$. In (16), $(I + E_L)$ and $(I + E_R)$ appear in a different side than in (4). It is easy to see that this does not change the first order error bounds. Notice that we have replaced the unit row scaling of $\widehat{L}_\omega$ with the scaling given by $(D')^{-1}$. We can do this because the condition number of the former matrix is not larger than a factor $\sqrt{r}$ times the condition number of the latter [21]. Note also that $\kappa((D')^{-1}\widehat{L}_\omega) = \kappa((D')^{-1}\widehat{L}_\omega\,(Q'_\omega)^T) = \kappa(Z + E_z) = \kappa(Z)(1 + O(\epsilon)\kappa(Z))$. Hence,

$$\|E_R\| = O(\epsilon\kappa(R')\kappa(Y)).$$

Substituting (16) into (14) leads to

$$(I + E_3)G(I + F_3) = Q'\overline{U}'\widehat{\Sigma}(\overline{V}')^T(Q'_\omega)^T,$$

where $I + E_3 = (I + \widetilde{E}_L)^{-1}(I + E_2)$ and $I + F_3 = (I + F_2)(I + \widetilde{E}_R)^{-1}$ for $\widetilde{E}_L = Q'E_L(Q')^T$ and $\widetilde{E}_R = Q'_\omega E_R(Q'_\omega)^T$. Clearly, $\|E_3\| = O(\epsilon\kappa(X))$ and $\|F_3\| = O(\epsilon\kappa(R')\kappa(Y))$.

Finally, it only remains to show that $\widehat{U} = \texttt{fl}(Q\widehat{U}_\omega)$ and $\widehat{V} = \texttt{fl}(\widehat{Q}_\omega\widehat{V}_\omega)$ differ from $Q'\overline{U}'$ and $Q'_\omega\overline{V}'$ by $O(\epsilon)$. We show it for $\widehat{U}$; the argument for $\widehat{V}$ is analogous. Using (12) and (15), we obtain $Q\widehat{U}_\omega = Q'\overline{U}' + O(\epsilon)$. Moreover, the standard error

analysis for matrix multiplication implies that $\|\widehat{U} - Q\widehat{U}_\omega\|_F \leq r^2\epsilon + O(\epsilon^2)$. The proof is concluded by observing that $\|\widehat{U} - Q'\overline{U}'\|_F \leq \|\widehat{U} - Q\widehat{U}_\omega\|_F + \|Q\widehat{U}_\omega - Q'\overline{U}'\|_F$. ∎

Multiplicative perturbation theory for the SVD applied to (10) yields relative error bounds of order $O(\epsilon\,\kappa(R')\,\max(\kappa(X),\kappa(Y)))$ on the singular values and of order $O(\epsilon\kappa(R')\max(\kappa(X),\kappa(Y)))$ divided by the relative gaps on the singular vectors. These are the bounds previously obtained in [4, Theorem 3.1]. The backward multiplicative error (10) in Theorem 3.1 for Algorithm 1 can be easily combined with the backward multiplicative error coming from computing a RRD, with errors (7), (8), to produce an overall multiplicative backward error similar to (10) [9, section 2.1]. Other more general forward errors in the computation of a RRD can be managed in a similar way.

4. The left-handed version. The backward error analysis in section 3 has been performed assuming that right-handed Jacobi is employed in step 4 of Algorithm 1. However, it has been observed that Algorithm 1 with *left-handed* Jacobi on L_ω is usually much faster. For instance, for rank-revealing decompositions coming from quasi-Cauchy matrices, the following differences in computational cost (using double precision arithmetic) have been reported in [3, p. 572]: 50 Jacobi sweeps if right-handed Jacobi is used in step 4 and no more than 8 sweeps (4.6 on average) for the left-handed version. In the numerical experiments presented in [9, section 6.2] for random 100×100 matrices in RRD form, the average number of sweeps in the right version doubles the number of sweeps in the left version.[4] A heuristic reason of this significant difference in computational cost is that the rows of L_ω are usually closer to being orthogonal than its columns; thus left-handed Jacobi is expected to converge faster (see [20, p. 988] for a more detailed explanation of the advantages of one version of one-sided Jacobi over the other depending on the scaling). These discrepancies in speed make it interesting to undertake a brief analysis of the multiplicative backward stability properties of Algorithm 1 using left-handed Jacobi in step 4. Before we begin, it should be noted that all these remarks may be modified by future improvements in one-sided Jacobi SVD algorithms. According to numerical tests conducted using a preliminary version of the fast and sophisticated right-handed Jacobi routine which is being developed by Drmač, right-handed Jacobi could be much faster than the usual plain implementation of left-handed Jacobi.

The error bounds for left-handed Jacobi on an invertible matrix $A \in \mathbb{R}^{n\times n}$ remain as in Theorem 2.1, at the prize of replacing the $O(\epsilon\kappa(A_N))$ with $O(\epsilon\gamma)$, where

$$\gamma = \max_{i=0,1,\ldots,q} \kappa(B_i). \tag{17}$$

Here, each B_i is the diagonal scaling with unit rows of the matrix $A_i = D_i\,B_i$ ($A_0 = A$) resulting from the action of the ith finite precision rotation along the process of left-handed Jacobi, and A_q is the first iterate satisfying the stopping criterion

$$\max_{i\neq j} \mathtt{fl}\left(\frac{|A_q(i,:)A_q(j,:)^T|}{\|A_q(i,:)\|\,\|A_q(j,:)\|}\right) \leq n\,\epsilon \qquad \text{for} \quad i \neq j. \tag{18}$$

To explain the origin of the additional factor γ, notice that, according to [6, Theorem 4.1], if A_i (resp., A_{i+1}) is the matrix obtained after the ith (resp., $(i+1)$th) finite

[4] Both in [3] and in [9] Algorithm 1 runs on square matrices and has been implemented without step 3. If step 3 is done with row pivoting, then right-handed Jacobi can improve its speed by more than one sweep, but this is not enough to wipe out the differences with the left-handed version.

precision rotation, then A_{i+1} can be written as

$$A_{i+1} = R_{i+1}(A_i + \delta A_i),$$

where R_{i+1} is an exact rotation and the backward error δA_i is such that $\|\delta B_i\| \leq 72\epsilon + O(\epsilon^2)$ for the row scaling $\delta A_i = D_i\, \delta B_i$, where D_i is the diagonal matrix with the row norms of A_i on the diagonal. Hence,

$$A_{i+1} = R_{i+1} A_i (I + E_i)$$

with $\|E_i\| = \|A_i^{-1}\delta A_i\| = \|B_i^{-1}\delta B_i\| \leq (72\epsilon + O(\epsilon^2))\kappa(B_i)$. Notice that replacing $\|B_i^{-1}\|$ with $\kappa(B_i)$ increases the bound at most by a factor $\sqrt{n}$.

Repeating the argument for all q rotations up to convergence, one obtains

$$A_q = (\widetilde{U}')^T A(I + \widetilde{E})$$

for an exact orthogonal matrix $\widetilde{U}'$ and a matrix $\widetilde{E}$ such that $\|\widetilde{E}\| \leq (72\epsilon + O(\epsilon^2))q\gamma$, with γ given by (17). The constant q in the previous error bound is pessimistic, and in fact with a finer implementation of left-handed Jacobi q can be replaced by $(s-1)p$, where s is the number of sweeps up to convergence, each of them implemented in p parallell steps [11].

Using the stopping criterion as in the end of the proof of Theorem 2.1 shows that if $\widehat{U}\widehat{\Sigma}\widehat{V}^T$ is the SVD computed by left-handed Jacobi on A with stopping criterion (18), then

$$A(I + \widetilde{E}_R) = \widetilde{U}'\widehat{\Sigma}\widetilde{V}'^T$$

for orthogonal matrices $\widetilde{U}'$, $\widetilde{V}'$ within a distance $O(\epsilon)$ of $\widehat{U}$, $\widehat{V}$, and

$$\|\widetilde{E}_R\| \leq 72\epsilon q\gamma + cn^2\epsilon + O(\epsilon^2) = O(\epsilon\gamma).$$

This last bound makes explicit the proviso needed in [6] to guarantee that one-sided Jacobi is able to compute the SVD with high relative accuracy for matrices of the form DB, where D is diagonal and B is well-conditioned: γ cannot be much larger than $\kappa(B)$.

Plugging these backward errors into the proof of Theorem 3.1, we obtain for the left-handed version of Algorithm 1 (i.e., the one using left-handed Jacobi in step 4) the backward error bound

$$(I + \widetilde{E})G(I + \widetilde{F}) = U'\widehat{\Sigma}V'^T,$$

where, as in Theorem 3.1, U' and V' have orthonormal columns,

$$\|U' - \widehat{U}\| = O(\epsilon), \quad \|V' - \widehat{V}\| = O(\epsilon)$$

for the computed matrices $\widehat{U}$, $\widehat{\Sigma}$, $\widehat{V}$, and the backward errors satisfy

$$\|\widetilde{E}\| = O(\epsilon\kappa(X)), \quad \|\widetilde{F}\| = O(\epsilon \max\{\gamma, \kappa(R')\kappa(Y)\}),$$

with γ being the constant defined in (17) for left-handed Jacobi on the matrix $\widehat{L}_\omega$ computed in step 3 of Algorithm 1. Therefore, the error bounds for this left-handed version of Algorithm 1 are larger than those for the right-handed one. Only if γ is of the order $O(\kappa(R')\kappa(Y))$ the same accuracy will be achieved. It is claimed in [6] that there is strong numerical evidence of $\gamma/\kappa(B_0) \approx 1$. This has also been observed in the numerical experiments done in [9]. Hence, it seems that the increase in speed of the left-handed version is not penalized by a loss of accuracy.

Acknowledgment. The authors thank Prof. Zlatko Drmač for providing the source code for his state-of-the-art implementation of the one-sided Jacobi SVD routine, and also for many illuminating discussions on one-sided Jacobi SVD algorithms.

REFERENCES

[1] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen, *LAPACK User's Guide*, 3rd ed., SIAM, Philadelphia, 1999.

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

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

[4] J. Demmel, M. Gu, S. Eisenstat, I. Slapničar, K. Veselić, and Z. Drmač, *Computing the singular value decomposition with high relative accuracy*, Linear Algebra Appl., 299 (1999), pp. 21–80.

[5] J. Demmel and W. Kahan, *Accurate singular values of bidiagonal matrices*, SIAM J. Sci. Stat. Comput., 11 (1990), pp. 873–912.

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

[7] J. Demmel and P. Koev, *Accurate SVDs of polynomial Vandermonde matrices involving orthonormal polynomials*, Linear Algebra Appl., to appear.

[8] J. Demmel and P. Koev, *Accurate SVDs of weakly diagonally dominant M-matrices*, Numer. Math., to appear.

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

[10] Z. Drmač, *Implementation of Jacobi rotations for accurate singular value computation in floating-point arithmetic*, SIAM J. Sci. Comput., 18 (1997), pp. 1200–1222.

[11] Z. Drmač, *Accurate computation of the product-induced singular value decomposition with applications*, SIAM J. Numer. Anal., 35 (1998), pp. 1969–1994.

[12] Z. Drmač, *A posteriori computation of the singular vectors in a preconditioned Jacobi SVD algorithm*, IMA J. Numer. Anal., 19 (1999), pp. 191–213.

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

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

[15] G. H. Golub and C. F. Van Loan, *Matrix Computations,* 3rd ed., The Johns Hopkins University Press, Baltimore, 1996.

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

[17] I. Ipsen, *Relative perturbation results for matrix eigenvalues and singular values*, Acta Numerica (1998), pp. 151–201.

[18] R.-C. Li, *Relative perturbation theory:* I. *Eigenvalue and singular value variations*, SIAM J. Matrix Anal. Appl., 19 (1998), pp. 956–982.

[19] R.-C. Li, *Relative perturbation theory:* II. *Eigenspace and singular subspace variations*, SIAM J. Matrix Anal. Appl., 20 (1999), pp. 471–492.

[20] R. Mathias, *Accurate eigensystem computations by Jacobi methods*, SIAM J. Matrix Anal. Appl., 16 (1995), pp. 977–1003.

[21] A. van der Sluis, *Condition numbers and equilibration of matrices*, Numer. Math., 14 (1969), pp. 14–23.

[22] G. W. Stewart and J.-G. Sun, *Matrix Perturbation Theory*, Academic Press, New York, 1990.