# Computing the Singular Value Decomposition with High Relative Accuracy

James Demmel[*], Ming Gu[†], Stanley Eisenstat[‡], Ivan Slapničar[§], Krešimir Veselić[¶], Zlatko Drmač[‖]

February 7, 1997

**Abstract**

We analyze when it is possible to compute the singular values and singular vectors of a matrix with high relative accuracy. This means that each computed singular value is guaranteed to have some correct digits, even if the singular values have widely varying magnitudes. This is in contrast to the absolute accuracy provided by conventional backward stable algorithms, which in general only guarantee correct digits in the singular values with large enough magnitudes. It is of interest to compute the tiniest singular values with several correct digits, because in some cases, such as finite element problems and quantum mechanics, it is the smallest singular values that have physical meaning, and should be determined accurately by the data. Many recent papers have identified special classes of matrices where high relative accuracy is possible, since it is not possible in general. The perturbation theory and algorithms for these matrix classes have been quite different, motivating us to seek a common perturbation theory and common algorithm. We provide these in this paper, and show that high relative accuracy is possible in many new cases as well. The briefest way to describe our results is that we can compute the SVD to high relative accuracy provided we can compute a "high accuracy" pivoted LDU decomposition. We provide many examples of matrix classes permitting such an LDU decomposition.

[*]Computer Science Division and Mathematics Dept., University of California, Berkeley, CA 94720 (`demmel@cs.berkeley.edu`). This material is based in part upon work supported by the Advanced Research Projects Agency contract No. DAAL03-91-C-0047 (via subcontract No. ORA4466.02 with the University of Tennessee), the Department of Energy grant No. DE-FG03-94ER25219, and contract No. W-31-109-Eng-38 (via subcontract Nos. 20552402 and 941322401 with Argonne National Laboratory), the National Science Foundation grant ASC-9313958, and NSF Infrastructure Grant Nos. CDA-8722788 and CDA-9401156.

[†]Mathematics Dept., UCLA, Los Angeles, CA 90095 (`mgu@math.ucla.edu`). Part of this work was done when the author was with the Department of Mathematics, University of California at Berkeley and Lawrence Berkeley National Laboratory, and was partially supported by Applied Mathematical Sciences Subprogram of the Office of Energy Research, U.S. Department of Energy under Contract DE-AC03-76SF00098.

[‡]Department of Computer Science, Yale University, P. O. Box 208285, New Haven, CT 06520-8285 (`eisenstat-stan@cs.yale.edu`). The research of this author was supported in part by NSF grant CCR-9400921.

[§]University of Split, Faculty of Electrical Engineering, Mechanical Engineering, and Naval Architecture, R. Boškovića b.b, 21000 Split, Croatia (`ivan.slapnicar@fesb.hr`). This work was supported by Croatian Ministry of Science and Technology under grant No. 037012.

[¶]Lehrgebiet Mathematische Physik, Fernuniversität Hagen, Postf. 940, 58084 Hagen, Germany (`Kresimir.Veselic@FernUni-Hagen.de`).

[‖]Department of Computer Science, Engineering Center ECOT 7–7, University of Colorado at Boulder, Boulder, CO 80309–0430 (`zlatko@boulder.colorado.edu`). This research was supported by National Science Foundation grants ACS–9357812 and ASC–9625912, Department of Energy grant DE–FG03–94ER25215, and the Intel Corporation.

# 1 Introduction

The *singular value decomposition (SVD)* of a real matrix $G$ is the factorization $G = U\Sigma V^T$ where $U$ and $V$ are orthogonal matrices and $\Sigma$ is nonnegative and diagonal. If $G$ is $m$-by-$n$, with $m \geq n$ (otherwise transpose $G$), then $U$ is $m$-by-$n$, $\Sigma = \mathrm{diag}(\sigma_1, ..., \sigma_n)$ with $\sigma_1 \geq \cdots \geq \sigma_n \geq 0$, and $V$ is $n$-by-$n$. We call the columns $u_i$ of $U = [u_1, ..., u_n]$ the *left singular vectors*, the columns $v_i$ of $V = [v_1, ..., v_n]$ the *right singular vectors*, and the $\sigma_i$ the *singular values*.

Our goal is to compute the SVD (i.e. the $u_i$, $v_i$ and $\sigma_i$) as accurately as the data deserves, using conventional floating point arithmetic. The phrase "as the data deserves" means that we assume that there is an unknown but bounded perturbation $\delta G$, and that we are given $\hat{G} = G + \delta G$ as input, not $G$ itself. Thus a properly posed problem includes an input matrix $\hat{G}$, and some information about how $\delta G$ is bounded. The inherent uncertainty in the data represented by the bound on $\delta G$ will limit the accuracy with which we can compute the SVD of $G$, independent of any additional errors introduced by the algorithms.

To explain the higher accuracy to which we aspire to compute the SVD, we will contrast it with the accuracy provided by conventional SVD algorithms, such as QR iteration, bisection and inverse iteration, or divide-and-conquer [12, 28, 31]. Their model of uncertainty asserts that $\delta G$ is bounded in norm, and that $\|\delta G\|/\|G\| \ll 1$ ($\|\cdot\|$ is the two-norm). This model of uncertainty is appropriate because roundoff error in these algorithms means that $\delta G$ typically satisfies $\|\delta G'\|/\|G\| = \Omega(\varepsilon)$ (i.e. at least order $\varepsilon$) where $\varepsilon$ is the *machine precision*, or maximum relative error in any floating point operation (barring over/underflow, which we ignore). Thus, including both input and roundoff error, these conventional algorithms only compute the SVD of $\hat{G} = G + \delta G$, where

$$\|\delta G\| \leq \eta \|G\| \text{ where } 0 \leq \eta \ll 1 \ . \tag{1}$$

This bound may be combined with *perturbation theorems* [43, 58, 36, 37] to derive the following conventional error bounds:

Let $\hat{U} = [\hat{u}_1, ..., \hat{u}_n]$, $\hat{\Sigma} = \mathrm{diag}(\hat{\sigma}_1, ..., \hat{\sigma}_n)$, and $\hat{V} = [\hat{v}_1, ..., \hat{v}_n]$ be the SVD of

$$\hat{G} = G + \delta G \quad \text{where } \|\delta G\| \leq \eta \|G\| \, . \tag{2}$$

Then the difference between the true and perturbed singular values is bounded by

$$|\sigma_i - \hat{\sigma}_i| \leq \eta \cdot \|G\| = \eta \cdot \sigma_1 \ . \tag{3}$$

Furthermore, the acute angle $\theta$ between the true and computed left singular vectors $u_i$ and $\hat{u}_i$ (or between right singular vectors $v_i$ and $\hat{v}_i$) is bounded by

$$\sin\theta \leq \frac{\eta}{\mathrm{abs\_gap}(i, G, \hat{G})} \tag{4}$$

provided the *absolute gap*

$$\mathrm{abs\_gap}(i, G, \hat{G}) \equiv \min_{j \neq i} |\sigma_i - \hat{\sigma}_j|/\sigma_1 \tag{5}$$

between $\sigma_i$ and the nearest other singular value is positive. (This is true when $m = n$; when $m > n$ then $\mathrm{abs\_gap}(i, G, \hat{G})$ for the left singular vectors is the minimum of the above expression and $\sigma_i/\sigma_1$.)

We call this accuracy provided by conventional algorithms *absolute accuracy*, to contrast it with the more stringent *relative accuracy* described in the next paragraph:

Let $\hat{U} = [\hat{u}_1, ..., \hat{u}_n]$, $\hat{\Sigma} = \mathrm{diag}(\hat{\sigma}_1, ..., \hat{\sigma}_n)$, and $\hat{V} = [\hat{v}_1, ..., \hat{v}_n]$ be the SVD of

$$\hat{G} = (I+E)G(I+F) \quad \text{where } \|E\| \le \eta_E \text{ and } \|F\| \le \eta_F \ . \tag{6}$$

We call $\hat{G}$ in (6) a *multiplicative perturbation* of $G$, to contrast it with the *additive perturbation* of $G$ in (2). Then we say the SVD of $\hat{G}$ approximates the SVD of $G$ *with relative accuracy* $\eta'$, *where* $\eta \equiv \max(\eta_E, \eta_F)$ and $\eta' = 2\eta + \eta^2$, since [24, Thm. 3.1] [27, Lemma 6.4.]

$$\frac{|\sigma_i - \hat{\sigma}_i|}{\sigma_i} \le \eta_E + \eta_F + \eta_E\eta_F \le 2\eta + \eta^2 = \eta' \tag{7}$$

and the acute angle $\theta$ between $u_i$ and $\hat{u}_i$ (or between $v_i$ and $\hat{v}_i$) satisfies [24, Thm 3.3]

$$\sin\theta \le \sqrt{2}\left(\frac{1+\eta'}{1-\eta'} \cdot \frac{\eta'}{\mathrm{rel\_gap}(i,G) - \eta'} + \eta\right) \tag{8}$$

provided that the *relative gap*

$$\mathrm{rel\_gap}(i,G) \equiv \min\left(\min_{j \ne i} \frac{|\sigma_i - \hat{\sigma}_j|}{\sigma_i}\ ,\ 2\right) \tag{9}$$

between $\sigma_i$ and the nearest other singular value is at least $\eta'$.

To make the difference between absolute and relative accuracy concrete, we consider bidiagonal matrices, which arise from computing the vibrational frequencies of a linear mass-spring system, as described in section 12.1. In particular, consider the 3-by-3 bidiagonal matrix

$$G = \begin{bmatrix} 1.2435 \cdot 10^{-9} & 5.8978 \cdot 10^{-9} & 0 \\ 0 & -2.0970 \cdot 10^{-8} & .92354 \\ 0 & 0 & .38350 \end{bmatrix}$$

which has singular values $\sigma_1 \approx 1$, $\sigma_2 \approx 10^{-8}$ and $\sigma_3 \approx 10^{-9}$. Suppose we perturb $G$ by multiplying each $g_{i,j}$ by a factor $1 + \epsilon_{i,j}$, where $|\epsilon_{i,j}| \le 10^{-6}$. As before, call the perturbed matrix $\hat{G} = G + \delta G$. Then we can only assert that $\|\delta G\| \lessapprox 10^{-6}$, and so apply absolute bounds (3) and (4) with $\eta = 10^{-6}$. In contrast, Theorem 8.1 below (as well as theorems in [3, 18, 16]) assert that we can write $G + \delta G = (I+E)G(I+F)$ where $E$ and $F$ are (diagonal) matrices of norm at most about $2.5 \cdot 10^{-6}$, so relative bounds (7) and (8) apply with $\eta = 2.5 \cdot 10^{-6}$. This leads to the perturbation bounds in Table 1. The relative error bounds this table guarantee that the two smaller singular values and their singular vectors are accurate to about 5 decimal digits, whereas the absolute error bounds guarantee no correct digits at all. Algorithms capable of computing the SVD of bidiagonal matrices with such high relative accuracy were published in [18, 16, 25].

Our interest in the notion of relative accuracy defined by bounds (7) and (8) arises for two reasons. First, there are a number of physical problems where the smallest singular values (or eigenvalues) are well-determined by the physical problem being modeled, and we need to compute them with some relative accuracy. For example, modes of vibration of finite element problems,

| Absolute Bounds | Relative Bounds |
| --- | --- |
| Singular Value Bounds | |
| $\hat{\sigma}_1 = 1 \pm 10^{-6}$<br>$\hat{\sigma}_2 = 10^{-8} \pm 10^{-6}$<br>$\hat{\sigma}_3 = 10^{-9} \pm 10^{-6}$ | $\hat{\sigma}_1 = 1 \cdot (1 \pm 5 \cdot 10^{-6})$<br>$\hat{\sigma}_2 = 10^{-8} \cdot (1 \pm 5 \cdot 10^{-6})$<br>$\hat{\sigma}_3 = 10^{-9} \cdot (1 \pm 5 \cdot 10^{-6})$ |
| Gaps | |
| $\text{abs\_gap}(1, G) \approx 1$<br>$\text{abs\_gap}(2, G) \approx .9 \cdot 10^{-8}$<br>$\text{abs\_gap}(3, G) \approx .9 \cdot 10^{-8}$ | $\text{rel\_gap}(1, G) \approx 1$<br>$\text{rel\_gap}(2, G) \approx .9$<br>$\text{rel\_gap}(3, G) \approx 2$ |
| Singular Vector Bounds | |
| $\theta(u_1, \hat{u}_1) \lessapprox 10^{-6}$<br>$\theta(u_2, \hat{u}_2) \lessapprox 1.1 \cdot 10^{2}$<br>$\theta(u_3, \hat{u}_3) \lessapprox 1.1 \cdot 10^{2}$ | $\theta(u_1, \hat{u}_1) \lessapprox 1.1 \cdot 10^{-5}$<br>$\theta(u_2, \hat{u}_2) \lessapprox 1.1 \cdot 10^{-5}$<br>$\theta(u_3, \hat{u}_3) \lessapprox 7.1 \cdot 10^{-6}$ |

Table 1: Absolute versus Relative Error Bounds

and energy levels in quantum mechanical systems fall in this class[1]. The second reason is that a large number of recent papers describe apparently unrelated classes of matrices $G$, and classes of perturbations $\delta G$, such that the SVDs of $G$ and $\hat{G} = G + \delta G$ agree to high relative accuracy, as described by bounds (7) and (8). Many of these papers also provide quite different algorithms that compute the SVD with these bounds, where $\eta$ is proportional to machine epsilon $\varepsilon$. These matrix classes include

1. bidiagonal matrices [18, 16, 25]
2. acyclic matrices [17] (see below for a definition)
3. scaled diagonally dominant matrices [3]
4. well-scalable symmetric positive definite matrices [19], and
5. certain well-scalable symmetric indefinite matrices [57, 49, 48].

Some of these results depended on the multiplicative perturbation theory stated above, and others did not. In other words, special techniques were used in each case.

In this paper we present a single perturbation theory that includes all the cases in the above list, as well as several new ones. We also provide an algorithm, which with some variations computes the SVD to high relative accuracy in all known cases.

Here is an outline of our results.

1. In section 2 we define a *rank-revealing decomposition (RRD)* of a matrix $G$ to be *any* representation of the form $G = XDY^T$, where $D$ is diagonal, $X$ and $Y$ have at least as many rows as columns, and $X$ and $Y$ are "well-conditioned". The SVD itself is such a representation

[1] In one example communicated to the authors, the smallest eigenvalue of a discretized Schrödinger operator was desired to several digits of relative accuracy, though the largest eigenvalue was $10^{70}$ times larger [26].

($X$ and $Y$ are perfectly conditioned), but there are many others, depending on how large a condition number for $X$ and $Y$ one will tolerate. For example the factorizations provided by rank-revealing QR [51, 4, 13, 14, 30, 33, 42] and Gaussian elimination with complete pivoting (GECP) or other pivoting [41, 44]. GECP (usually) provides an RRD since the unit lower and upper triangular factors $L$ and $U$ in $G = P_r LDU P_c$ have offdiagonal entries bounded by 1 in absolute value (here $P_r$ and $P_c$ are permutations[2]).

We then show that if we perturb the RRD $G = XDY^T$ to get $\hat{G} = \hat{X}\hat{D}\hat{Y}^T$, where

$$\begin{aligned} \hat{X} &= X + \delta X \quad \text{where} \quad \frac{\|\delta X\|}{\|X\|} \le \epsilon \\ \hat{D} &= D + \delta D \quad \text{where} \quad \delta D \text{ is diagonal and } \frac{|\delta D_{ii}|}{|D_{ii}|} \le \epsilon \\ \hat{Y} &= Y + \delta Y \quad \text{where} \quad \frac{\|\delta Y\|}{\|Y\|} \le \epsilon \end{aligned} \tag{10}$$

then relative bounds (7) and (8) hold with $\eta = O(\epsilon \max(\kappa(X), \kappa(Y)))$, where $\kappa(Z) = \frac{\sigma_{\max}(Z)}{\sigma_{\min}(Z)}$ is the condition number of $Z$; see Theorem 2.1 in section 2. This implies that an approximate RRD (in the sense of (10) determines the SVD to high relative accuracy $\eta$.

2. In section 3, we show that given any RRD $G = XDY^T$, one can compute the SVD of $G$ with relative error bounds (7) and (8), where $\eta = O(\varepsilon \max(\kappa(X), \kappa(Y)))$. We actually have several algorithms for this, of slightly varying complexity and accuracy. The algorithm we present in detail (Algorithm 3.1 in section 3) uses only QR decomposition with pivoting, matrix multiplication (twice), and one-sided Jacobi as its ingredients. This implies that any method for computing any accurate RRD of $G$ (in the sense of (10)) permits us to compute the SVD of $G$ to high relative accuracy.

3. It remains to ask which classes of matrices permit accurate RRDs to be computed. We concentrate on the RRD provided by GECP, since this works so widely. These classes depend on two different characterizations of Gaussian elimination. The first characterization expresses the entries of $L$, $D$ and $U$ as entries of *Schur complements*, i.e. expressions of the form $S = G_{22} - G_{21}G_{11}^{-1}G_{12}$. By imposing conditions on (scaled) condition numbers of submatrices like $G_{11}$, we can guarantee that $S$ is computed accurately. The second characterization expresses the entries of $L$, $D$ and $U$ as quotients of minors of $G$. By imposing conditions on the sparsity, signs, or algebraic relations among entries of $G$, we can guarantee that all minors are determined accurately. Sometimes straightforward GECP does not compute the LDU factors sufficiently accurately, in which cases we show how to modify it to do so. We discuss these classes in more detail below.

4. In section 4 we discuss conditions on condition numbers of (scaled) submatrices of $G$ that guarantee that Schur complements, and so entries of an LDU factorization, can be accurately computed. In particular, suppose we write $G = D_1 B D_2$ where $D_1$ and $D_2$ are diagonal. We consider the case where $B$ is well-conditioned, and $D_1$ and $D_2$ "contain" any ill-conditioning of $G$. We provide conditions on $B$ that guarantee that GECP is accurate independently of

[2] If we ignore factors depending only on dimension, then GECP always computes an RRD; this depends on how loosely we interpret the meaning of "well-conditioned" in our definition of RRD.

$D_1$ and $D_2$. Since $D_1$, $B$ and $D_2$ are not always known, we also provide a computable a posteriori error bound that depends only on the computed $LDU$ factors computed by GECP.

This work generalizes earlier work where $B$ was symmetric positive definite, and $D_1 = D_2$ [19]. In this simpler case, it was enough for $B$ to be well-conditioned to get the singular values to high relative accuracy. In contrast, the general case considered here requires *all* submatrices of $B$ to be well-conditioned, in order that high relative accuracy be attained independent of $D_1$ and $D_2$. Since an $n$-by-$n$ matrix has $O(4^n)$ submatrices, this is a lot of conditions to satisfy. But our a posteriori error bound reduces this possibly exponential cost back to $O(n^3)$ by measuring the condition numbers of just the relevant submatrices.

5. Section 5 outlines the combinatorial and algebraic conditions that guarantee that the LDU factors of $G$ are determined to high relative accuracy. We rely on the fact that entries of $L$, $D$ and $U$ are quotients of minors of $G$, and ask when there are formulas for these minors that can be evaluated with high relative accuracy. For example, a formula containing only multiplication, division, and addition of quantities of like signs, is evaluatable to high relative accuracy; the only dangerous operation of subtractive cancellation is excluded. But we can in fact permit cancellation, as long as the operands are *input floating point data*, i.e. can be treated as exact values stored in the machine. If cancellation $a - b$ does occur in such a case, the result is *exact*[3].

   To illustrate the conditions we impose, consider computing the determinant $D = g_{11}g_{22} - g_{12}g_{21}$ of the 2-by-2 matrix $G$ with entries $g_{ij}$. For $D$ to be determined to high relative accuracy, independent of the magnitudes of the $g_{ij}$, we could impose conditions on the *sparsity pattern* of $G$, for example insisting that at least one entry of $G$ be exactly zero; in this case either $D = g_{11}g_{22}$, $D = -g_{12}g_{21}$, or $D = 0$, in all of which cases $D$ is determined to high relative accuracy. Or we could impose conditions on the *sign pattern* of $G$, for example insisting that $g_{12} \leq 0$ and otherwise $g_{ij} \geq 0$; in this case computing $D = g_{11}g_{22} - g_{12}g_{21}$ involves adding positive numbers, and so no cancellation.

   It turns out that there are simple necessary and sufficient conditions on the sparsity and sign patterns, that guarantee that all $LDU$ factors can be computed to high relative accuracy. Section 6 discusses the sparsity pattern by itself; the condition is that that *graph of $G$* be acyclic [17]. Acyclic matrices include bidiagonal matrices, for example. Section 7 discusses sign and sparsity patterns; the condition is that $G$ be *total signed compound (TSC)* [11]. TSC matrices include acyclic matrices, tridiagonal and "arrow" matrices with the sign patterns

$$\begin{bmatrix} + & + & & & \\ + & - & + & & \\ & + & + & + & \\ & & + & - & + \\ & & & + & + \end{bmatrix} \text{ and } \begin{bmatrix} + & + & + & + & + \\ + & - & & & \\ + & & - & & \\ + & & & - & \\ + & & & & - \end{bmatrix} . \tag{11}$$

   and many others.

   It turns out that conventional GECP does *not* preserve the high relative accuracy inherently available, but we show how to modify GECP to regain high relative accuracy. Unfortunately,

[3]This excludes machines with sloppy floating point arithmetic, like the Cray T90, and its predecessors and emulators.

this modification can cost as much as $O(n^4)$ for TSC matrices; finding an $O(n^3)$ implementation is an open problem.

6. Section 8 considers *diagonally scaled totally unimodular (DSTU) matrices*. These include acyclic matrices as special cases, as well as certain finite element problems considered in section 12. A *totally unimodular (TU) matrix* is an integer matrix $Z$ each of whose minors is $-1$, 0 or $+1$, and a DSTU matrix is of the form $D_1 Z D_2$, where $D_1$ and $D_2$ are diagonal and $Z$ is TU. DSTU matrices include the reduced node-arc incidence matrices analyzed in [55].

7. Section 9 discusses *Cauchy matrices*, i.e. matrices whose entries are $C_{i,j} = 1/(x_i + y_j)$, where $x_1, ..., x_n$ and $y_1, ..., y_m$ are given data. There is a classical formula for $\det(C)$ that satisfies our conditions above for being evaluatable to high relative accuracy. The cost of the modified version of GECP using this formula is $O(n^5)$, so faster algorithms would be welcome.

8. Section 10 discusses *totally positive (TP) matrices*, i.e. matrices all of whose minors are nonnegative. The Hilbert matrix is an example (it is also Cauchy), and TP matrices arise elsewhere frequently in applied mathematics [35]. There are many way to parameterize TP matrices; the parameters $x_i$ and $y_j$ above for a Cauchy matrix is one example of many. The existence of high relative accuracy formulas for minors depends on choosing the right parameterization. There turns out to be a systematic way to develop good parameterizations, and corresponding high accuracy formulas for minors, for all TP matrices. Unfortunately, the costs of these formulas are sometimes exponential in $n$, and we do not know if we can do better.

9. Section 11 discusses which other linear algebra problems besides the SVD can be solved to high accuracy, given the combinatorial and algebraic conditions described in earlier sections. Since solutions of linear systems, and some aspects of least squares problems, can be expressed in terms of minors, it is no surprise that a matrix whose minors are determined accurately also determines its inverse accurately.

10. *Finite element matrices*, which are discussed in Section 12, arise from many problems where we want to compute the vibrational frequencies of some physical system. Usually the lowest frequencies (eigenvalues) are of physical interest, so we want to compute them accurately. The most natural formulation usually leads to a generalized eigenvalue problem of the form $Kx = \lambda Mx$, where $M$ is the *mass matrix*, and $K$ is the *stiffness matrix*. Typically we write $K = Z_K^T D_K Z_K$ where $Z_K$ is the *incidence matrix* or *assembly matrix*, and $D_K$ is the (block)diagonal matrix of individual element stiffnesses. We may similarly write $M = Z_M^T D_M Z_M$.

    We will see that we can sometimes reduce the eigenproblem $K - \lambda M$ to the SVD of a single matrix $G = D_1 B D_2$, where $D_1$ and $D_2$ are diagonal, and depend only on the *material properties* (masses and spring constants in $D_K$ and $D_M$), and $B$ depends only on the *geometry and meshing* of the finite element model (in $Z_K$ and $Z_M$). The relative accuracy attainable by our algorithms will depend *only* on $B$, i.e. on the geometry and meshing, and be independent of the material properties in $D_1$ and $D_2$. A similar analysis of linear systems arising in finite element problems appears in [46, 55, 54].

    To reduce the length of this paper, we will only present one example in detail, a *linear mass-spring system* consisting of masses that can move in one dimension only. In this simple case the relative accuracy depends only on the relative accuracy with which the individual masses and spring constants are known. More complicated finite element problems will be considered in a future paper.

The last section, section 13, lists open problems. Finally, we note that the sequence of initials of the last names of the authors, DGESVD, is the name of the most accurate LAPACK [1] routine currently available for the dense SVD (which only provides high absolute accuracy).

# 2 Rank Revealing Decompositions (RRDs)

We repeat the following definition from the introduction:

**Definition 2.1** Let $G$ be $m$-by-$n$ with $m \geq n$. Let $X$ be $m$-by-$r$, $D$ be $r$-by-$r$, and $Y$ be $n$-by-$r$, where $r \leq \min(m,n)$. Then $G = XDY^T$ is a *rank-revealing decomposition (RRD)* of $G$ if $X$ and $Y$ are well-conditioned, and $D$ is diagonal and nonsingular.

The SVD itself is such a decomposition, with $X$ and $Y$ optimally conditioned, i.e. orthogonal. But there are many other RRDs as well, most importantly the decomposition (usually) provided by Gaussian elimination with complete pivoting (GECP). The goal of this section and the next is to show that any RRD is as good as the SVD, in the sense that small changes in the factors of the RRD determine the SVD to high relative accuracy, and that there are efficient algorithms for computing the SVD this accurately, given any RRD.

**Theorem 2.1** Let $G = XDY^T$ be an RRD with SVD $G = U\Sigma V^T$, and let $\hat{G} = \hat{X}\hat{D}\hat{Y}^T$ with SVD $\hat{G} = \hat{U}\hat{\Sigma}\hat{V}^T$, where $\hat{X}$, $\hat{D}$ and $\hat{Y}$ are defined as in equation (10):

$$\begin{aligned}
\hat{X} &= X + \delta X \quad \text{where} \quad \frac{\|\delta X\|}{\|X\|} \leq \epsilon \\
\hat{D} &= D + \delta D \quad \text{where} \quad \delta D \text{ is diagonal and } \frac{|\delta D_{ii}|}{|D_{ii}|} \leq \epsilon \\
\hat{Y} &= Y + \delta Y \quad \text{where} \quad \frac{\|\delta Y\|}{\|Y\|} \leq \epsilon
\end{aligned}$$

where $0 \leq \epsilon < 1$. Let $\eta = \epsilon(2+\epsilon)\max(\kappa(X), \kappa(Y))$ and $\eta' = 2\eta + \eta^2$, where $\kappa(Z) = \sigma_{\max}(Z)/\sigma_{\min}(Z)$ is the condition number of $Z$. Then the difference between the singular values of $G$ and $\hat{G}$ is bounded as follows

$$\frac{|\sigma_i - \hat{\sigma}_i|}{\sigma_i} \leq \eta' \ . \tag{12}$$

Furthermore, the angle $\theta$ between $u_i$ and $\hat{u}_i$ (or between $v_i$ and $\hat{v}_i$) is bounded by

$$\sin\theta \leq \sqrt{2}\left(\frac{1+\eta'}{1-\eta'} \cdot \frac{\eta'}{\mathrm{rel\_gap}(i,G) - \eta'} + \eta\right) \tag{13}$$

provided that the *relative gap*

$$\mathrm{rel\_gap}(i,G) \equiv \min\left(\min_{j \neq i} \frac{|\sigma_i - \hat{\sigma}_j|}{\sigma_i}\ ,\ 2\right)$$

is at least $\eta'$.

The proof is simple. We use the *multiplicative perturbation theorems* stated in the introduction, which we repeat here.

**Theorem 2.2** *[24, Thm. 3.1]* Suppose $\hat{G} = (I+E)G(I+F)$, with $\|E\| = \eta_E$ and $\|F\| = \eta_F$. Let $\eta = \max(\eta_E, \eta_F)$ and $\eta' = 2\eta + \eta^2$. Then

$$\frac{|\sigma_i - \hat{\sigma}_i|}{\sigma_i} \leq \eta_E + \eta_F + \eta_E \eta_F \leq \eta' \ . \tag{14}$$

**Theorem 2.3** *[37, Thm. 3.5]* Suppose $\hat{G} = (I+E)G(I+F)$, where $\|E\| \leq \eta_E$ and $\|F\| \leq \eta_F$. Let $\eta = \max(\eta_E, \eta_F)$ and $\eta' = 2\eta + \eta^2$. Then the acute angle $\theta$ between $u_i$ and $\hat{u}_i$ (or between $v_i$ and $\hat{v}_i$) is bounded by

$$\sin\theta \leq \sqrt{2}\left(\frac{1+\eta'}{1-\eta'} \cdot \frac{\eta'}{\mathrm{rel\_gap}(i,G) - \eta'} + \eta\right) \tag{15}$$

provided that the *relative gap*

$$\mathrm{rel\_gap}(i,G) \equiv \min\left(\min_{j \neq i} \frac{|\sigma_i - \hat{\sigma}_j|}{\sigma_i} \ , \ 2\right)$$

between $\sigma_i$ and the nearest other singular value is at least $\eta'$.

The paper [37] includes similar perturbation theorems for singular subspaces, not just singular vectors. These are useful when several singular values form a tight cluster, and bounds are desired for the space spanned by their corresponding singular vectors.

*Proof of Theorem 2.1:* We write $\hat{G}$ in the form $(I+E)G(I+F)$: First write

$$\begin{aligned}
\hat{G} &= \hat{X}\hat{D}\hat{Y}^T \\
&= (X + \delta X)\hat{D}\hat{Y}^T \\
&= (I + \delta X X^+)X\hat{D}\hat{Y}^T \\
&\phantom{=}\ \text{where } X^+ \text{ is the Moore} - \text{Penrose pseudoinverse of } X \\
&= (I+E)X\hat{D}\hat{Y}^T \\
&\phantom{=}\ \text{where } E = \delta X X^+ \ .
\end{aligned}$$

Note that $\|E\| \leq \|\delta X\| \cdot \|X^+\| \leq \epsilon \|X\| \cdot \|X^+\| = \epsilon\kappa(X)$. Now we apply the same technique to the other two factors $\hat{D}$ and $\hat{Y}$. Note that $\hat{D} = D(I + D^{-1}\delta D) \equiv D(I+W)$, where $W$ is diagonal with norm bounded by $\epsilon$. Then

$$\begin{aligned}
\hat{G} &= (I+E)X\hat{D}\hat{Y}^T \\
&= (I+E)XD(I+W)(Y+\delta Y)^T \\
&= (I+E)XD(Y + YW + \delta Y(I+W))^T \\
&= (I+E)XD((I + [YW + \delta Y(I+W)]Y^+)Y)^T \\
&= (I+E)XDY^T(I + \{[YW + \delta Y(I+W)]Y^+\}^T) \\
&\equiv (I+E)XDY^T(I+F) \\
&= (I+E)G(I+F)
\end{aligned}$$

where $\|F\| \leq \kappa(Y)(2\epsilon + \epsilon^2)$. Applying Theorems 2.2 and 2.3 to $\hat{G} = (I+E)G(I+F)$ yields the result. $\square$

## 3 Computing the SVD from a Rank Revealing Decomposition

We present an algorithm for computing the SVD to high relative accuracy from an RRD $G = XDY^T$. After presenting Algorithm 3.1 in detail, we discuss some other algorithms for this problem briefly. None of our algorithms yet incorporates all known tricks for accelerating high accuracy algorithms [48, 39, 21]. Our goal here is simplicity and accuracy. A future paper will address speed issues.

**Algorithm 3.1** Computing the SVD $G = U\Sigma V^T$ given an RRD $G = XDY^T$.

(1) Perform QR factorization with pivoting on $XD$ to get $XD = QRP$, where $P$ is a permutation. Thus $G = QRPY^T$.
(2) Multiply to get $W = RPY^T$. This must be *conventional* matrix multiplication, e.g. Strassen's method [32] may not be used. Thus $G = QW$.
(3) Compute the SVD of $W = \bar{U}\Sigma V^T$ using one-sided Jacobi [19]. Thus $G = Q\bar{U}\Sigma V^T$.
(4) Multiply $U = Q\bar{U}$. Thus $G = U\Sigma V^T$ is the desired SVD.

**Remark.** Algorithm 3.1 actually computes just the nonzero singular values of $G$ and their corresponding singular vectors.

**Theorem 3.1** Let $D'$ be a diagonal matrix, chosen so that $R' = D'^{-1}R$ is as well conditioned as possible. We can always choose $D'$ so that $\kappa(R')$ is bounded by $O(2^n)$, and it is usually much smaller. Then in floating point arithmetic with machine precision $\varepsilon$, Algorithm 3.1 computes the SVD of $G$ with relative accuracy $\eta = O(\varepsilon\kappa(R') \cdot \max(\kappa(X), \kappa(Y)))$.

**Remark.** The factor $\kappa(R')$ in the error bound depends on how well the pivoting during the QR decomposition of $XD$ "reveals the rank" of $XD$. The bound $O(2^n)$ comes from the standard column-pivoting algorithm [28] and choosing $D'_{ii} = R_{ii}$, but better alternatives are available [51, 4, 13, 14, 30, 33, 42]. For example, Gu has a pivoting scheme that reduces $O(2^n)$ to $O(n^{1+(1/4)\log_2 n})$, analogous to the pivot growth bound for GECP. See also [41].

*Proof:* We proceed through the algorithm line by line, showing that the backward error introduced by every step but (3) is of the form $(I + E)G(I + F)$. The one-sided Jacobi algorithm in step (3) is described in [19, Alg. 4.1], and was shown to possess high relative accuracy when applied to matrices like $W$, which we will see is the product of a diagonal matrix $D'$ and a well-conditioned matrix $R'PY^T$ (modulo roundoff). The algorithm in step (3) is essentially the version of the one-sided Jacobi algorithm of Rutishauser in [59], but with a more stringent stopping criterion. (Later, more elegant proofs by Drmač [22] and Mathias [39] also use the fact that errors during one-sided Jacobi are of the form $(I+E)G(I+F)$, so that the entire error analysis propagates errors "multiplicatively" rather than "additively".)

Step (1) of Algorithm 3.1 may be written

$$\begin{aligned} G &= XDY^T \\ &= (QRP + E')Y^T \end{aligned}$$

where $Q$, $R$ and $P$ are the computed results from step (1), and $E'$ is the backward error. Since QR operates on columns of the matrix, it is easy to see that we can write

$$QRP = XD - E' = (X + \delta X)D$$

where for all $i$, column $i$ of $\delta X$ has norm bounded by $O(\varepsilon)$ times the norm of column $i$ of $X$. Thus

$$QRP = (I + \delta X X^+) X D = (I + E) X D$$

where $\|E\| \leq O(\varepsilon)\kappa(X)$. In other words, we may write

$$G = (I + E)^{-1} Q R P Y^T \quad .$$

Continuing with the algorithm, in Step (2) we multiply $RPY^T$ to get the computed product $W$, which satisfies $W + \delta W = RPY^T$, where $\delta W$ is the roundoff error. Since $P$ is a permutation, the order in which we perform the two multiplications to form $W$ does not matter. Since $R = D'R'$ where $D'$ is diagonal, and we use conventional matrix multiplication, we can bound $\delta W$ rowwise as follows ($e_i^T$ denotes the $i$-th row of the identity matrix):

$$\begin{aligned} \|e_i^T \delta W\| &\leq O(\varepsilon)\|e_i^T R\| \cdot \|Y^T\| \\ &= O(\varepsilon)|D'_{ii}| \cdot \|e_i^T R'\| \cdot \|Y^T\| \quad . \end{aligned}$$

Defining $Z = D'^{-1} W$, $\delta Z = D'^{-1} \delta W$, and letting $Z^+$ be the pseudoinverse of $Z$ we get

$$\|(Z + \delta Z)^+\| \leq \|R'^{-1}\| \cdot \|Y^+\| \quad ,$$

$$\|\delta Z\| = O(\varepsilon)\|R'\| \cdot \|Y^T\|$$

and

$$\begin{aligned} W + \delta W &= D'(Z + \delta Z) \\ &= D'Z(I + Z^+ \delta Z) \\ &= W(I + F') \end{aligned}$$

where

$$\begin{aligned} \|F'\| &\leq \|Z^+\| \cdot \|\delta Z\| \\ &\approx \|(Z + \delta Z)^+\| \cdot \|\delta Z\| \\ &\leq O(\varepsilon) \cdot \kappa(R') \cdot \kappa(Y) \quad . \end{aligned}$$

Altogether then, after Step (2), we have

$$G = (I + E)^{-1} Q W (I + F') \quad .$$

Next, in Step (3), we use one-sided Jacobi to compute the SVD of $W = D'Z$. We let $D'Z = \bar{U}'\Sigma' V'^T$ be the exact SVD, and let $\bar{U} = \bar{U}' - \delta\bar{U}$, $\Sigma = \Sigma' - \delta\Sigma$ and $V = V' - \delta V$ be the computed quantities returned by one-sided Jacobi. In [19] it is shown that to high relative accuracy $O(\varepsilon\kappa(Z)) = O(\varepsilon\kappa(R')\kappa(Y))$, $\bar{U}\Sigma V^T$ is the SVD of $D'Z$.

Continuing with the algorithm, we write

$$\begin{aligned} G &= (I + E)^{-1} Q (\bar{U}'\Sigma' V'^T)(I + F') \\ &= (I + E)^{-1} Q (\bar{U} + \delta\bar{U})(\Sigma + \delta\Sigma)(V + \delta V)^T (I + F') \\ &\quad \text{where } \bar{U},\ \Sigma, \text{ and } V \text{ are computed in Step (3) with errors } \delta\bar{U},\ \delta\Sigma, \text{ and } \delta V \\ &= (I + E)^{-1} (Q\bar{U} + Q\delta\bar{U})(\Sigma + \delta\Sigma)(V + \delta V)^T (I + F') \\ &= (I + E)^{-1} (U + \delta U + Q\delta\bar{U})(\Sigma + \delta\Sigma)(V + \delta V)^T (I + F') \\ &\quad \text{where } U \text{ is the computed product in Step (4) and } \delta U \text{ is the roundoff error.} \end{aligned}$$

Altogether, we get

$$(I+E)G(I+F) = (Q\bar{U}')\Sigma' V'^T = (U + Q\delta\bar{U} + \delta U)(\Sigma + \delta\Sigma)(V + \delta V)^T$$

where $I + F = (I + F')^{-1}$, so $\|E\| = O(\varepsilon\kappa(X))$ and $\|F\| \approx \|F'\| = O(\varepsilon\kappa(R')\kappa(Y))$. In other words, $(Q\bar{U}')\Sigma' V'^T$ is the true SVD of almost the right matrix $((I+E)G(I+F))$, and the computed SVD $U\Sigma V^T$ is almost the right SVD of almost the right matrix. (The fact that Q is not quite orthogonal does not matter here, since its nonorthogonality could be absorbed in the $I+E$ factor.)

Now apply multiplicative perturbation theorem 2.2 to see that the relative error in the singular values $\Sigma + \delta\Sigma$ is bounded by $O(\varepsilon \max(\kappa(X), \kappa(R')\kappa(Y)))$. From the analysis in [19], the relative difference between $\Sigma + \delta\Sigma$ and the actual computed output $\Sigma$ is also $O(\varepsilon\kappa(Z)) = O(\varepsilon\kappa(R')\kappa(Y))$. This proves that the relative error in the computed singular values is bounded by $O(\varepsilon \max(\kappa(X), \kappa(R')\kappa(Y)))$ as desired.

Finally, we consider the singular vectors. Multiplicative perturbation theorem 2.3 bounds the difference between the singular vectors of $G$ and those of $(I+E)G(I+F)$, namely the columns of $Q\bar{U}'$ and $V' = V + \delta V$, by $O(\varepsilon \max(\kappa(X), \kappa(R')\kappa(Y)))$ over the relative gaps. The analysis in [19] further bounds the the difference between the columns of $V + \delta V$ and the actual computed output $V$ by $O(\varepsilon\kappa(Z)) = O(\varepsilon\kappa(R')\kappa(Y))$ over the relative gaps, yielding the desired error bound for the right singular vectors.

For the left singular vectors, we introduced errors in steps (4) and (5), which we express as

$$U = Q\bar{U}' - Q\delta\bar{U} - \delta U$$

where $Q\bar{U}'$ are the true singular vectors of $(I+E)G(I+F)$. As with the left singular vectors, the analysis in [19] shows that each column of $\delta\bar{U}$ is bounded in norm by $O(\varepsilon\kappa(R')\kappa(Y))$ over the appropriate relative gap. Multiplying by $Q$ and adding $\delta U$ (which is bounded in norm by $O(\varepsilon)$) does not change this norm bound. Therefore, the errors in the columns of $U$ are bounded the same way. This yields the final desired error bound. □

Three other algorithms deserve mention. The first algorithm we discovered for this problem was based on an algorithm of Veselić and Slapničar. Assuming without loss of generality that $D_{ii} > 0$, we can write

$$\begin{aligned}
\begin{bmatrix} 0 & G \\ G^T & 0 \end{bmatrix} &= \begin{bmatrix} 0 & XDY^T \\ YDX^T & 0 \end{bmatrix} \\
&= \left\{ \frac{1}{\sqrt{2}} \cdot \begin{bmatrix} X & X \\ Y & -Y \end{bmatrix} \cdot \begin{bmatrix} D^{1/2} & 0 \\ 0 & D^{1/2} \end{bmatrix} \right\} \times \begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix} \times \\
&\quad \left\{ \frac{1}{\sqrt{2}} \cdot \begin{bmatrix} X & X \\ Y & -Y \end{bmatrix} \cdot \begin{bmatrix} D^{1/2} & 0 \\ 0 & D^{1/2} \end{bmatrix} \right\}^T \\
&\equiv Z \times J \times Z^T
\end{aligned}$$

By construction, the columns of $Z$ can be scaled so that its resulting condition number $\kappa(Z\tilde{D})$ is no larger than $\max(\kappa(X), \kappa(Y))$. Thus, we can apply Veselić's implicit J-orthogonal Jacobi algorithm [56] to compute the positive eigenvalues of $ZJZ^T$ (and their eigenvectors), which are the singular values of $G$ (and their singular vectors). The relative error in this algorithm is $O(\varepsilon\kappa(Z\tilde{D}))$ as desired, see Slapničar [48]. Unlike Algorithm 3.1, this algorithm has no $\kappa(R')$ factor in the error bound, but it is likely to be slower.

A very similar algorithm appeared in [45], which essentially applied a Jacobi-like iteration to the pencil

$$F^T F - \lambda \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix}, \text{ where } F = \begin{bmatrix} 0 & XD^{1/2} \\ YD^{1/2} & 0 \end{bmatrix} .$$

Finally, another algorithm appeared in [21, 22].

### 3.1 Numerical Experiments

In this section, we present results of numerical experiments with Algorithm 3.1, assuming we that are given an RRD $G = XDY^T$. We used Sun FORTRAN on a Sun SPARC 20 Workstation, with IEEE arithmetic. Our single precision procedure, `SGEPSV`, is implemented as follows. In step (1) we compute the QR factorization using LAPACK's [1] `SGEQPF` procedure, which does QR decomposition with column pivoting. Steps (2) and (4) are implemented using calls to the BLAS 3 [20] procedure `STRMM`, where we are careful to use an `STRMM` based on conventional matrix multiplication rather than Strassen's method, as required by the error analysis in Theorem 3.1. Step (3) has several possible implementations; we use the right-handed Jacobi scheme, i.e. the matrix $V$ is the accumulated product of Jacobi rotations. Since the dimension $r$ of $D$ is less than the number of columns $n$ of $G$, we save time by first computing the LQ factorization of $W$ and apply one-sided Jacobi to $L$. (We note that Algorithm 3.1 has a dual formulation that interchanges the roles of $X$ and $Y$. An optimized version would choose between versions depending on the sizes of the dimensions $m$, $r$ and $n$, but we will not pursue this here.)

We also use double precision versions of our routines, which have names beginning with `D` instead of `S`.

This set of experiments was designed to confirm the error analysis of Algorithm 3.1. We did this by constructing a set of $X$, $D$ and $Y$ with known condition numbers, computing the SVD of $XDY^T$ using both single precision and double precision (note that $G = XDY^T$ is never formed explicitly), and seeing whether the differences between the single precision and double precision singular values satisfied the error bound in Theorem 3.1 (they did). We also monitored the size of the $\kappa(R')$ term in the error analysis, and confirmed that it never grew larger than $O(100)$.

More precisely, here is what we did. We generated test triples $(X, D, Y)$ with dimensions $m = 200$, $r = 100$ and $n = 150$, and with specified values of $\kappa(X)$, $\kappa(D)$ and $\kappa(Y)$. In addition, $X$ and $Y$ had columns with unit 2-norm. The specified condition numbers were chosen to be $\kappa(X) = 10^i$ for $i = 2, 3, ..., 6$, $\kappa(D) = 10^j$ for $j = 2, 4, ..., 16$, and $\kappa(Y) = 10^k$ for $k = 2, 3, ..., 6$. $X$, $D$ and $Y$ were computed by the LAPACK [1] test matrix generator `DLATM1`, with their actual singular value distribution controlled by the parameter `MODE`. Two sets of modes where used, $(5, 4, -5)$ and $(3, -4, 5)$ for $(X, D, Y)$, respectively. Finally, for each of the $5 \cdot 8 \cdot 5 = 200$ values of $(i, j, k)$, and each of the 2 `MODE` settings, 4 random triplets $(X, D, Y)$ were generated, for $200 \cdot 2 \cdot 4 = 1600$ tests in all.

For each test triplet, we computed the singular values in single (yielding $\sigma_{S,i}$) and in double (yielding $\sigma_{D,i}$), and computed the error measure

$$\epsilon(G) = \frac{\max_i \frac{|\sigma_{S,i} - \sigma_{D,i}|}{\sigma_{D,i}}}{\max\{\kappa(X), \kappa(Y)\}} . \tag{16}$$

By Theorem 3.1, this ratio can be as large $O(\varepsilon_S \cdot \kappa(R'))$, where $\varepsilon_S = 2^{-24} \approx 6 \cdot 10^{-8}$, and $\kappa(R')$ should be $O(1)$. In fact, $\epsilon(G)$ never exceeded $6.1 \cdot 10^{-8}$ in all 1600 test cases. Furthermore, by

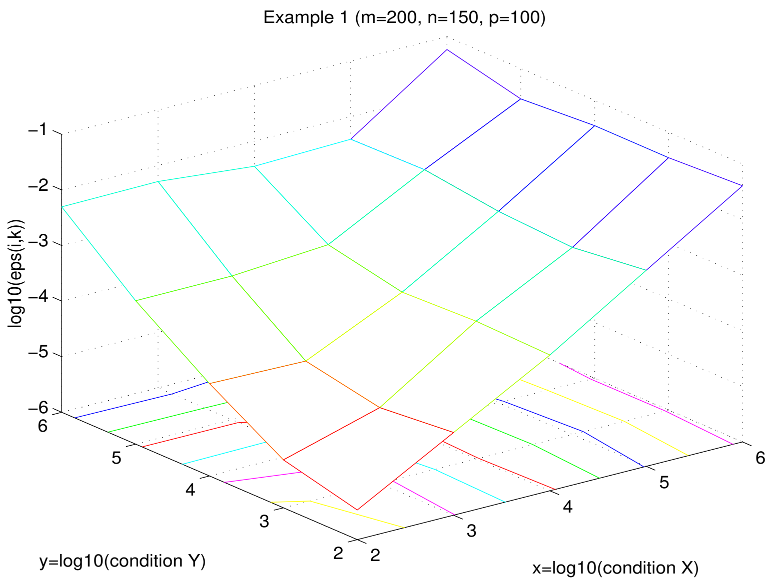



Figure 1: The values of of $\log_{10} \epsilon(i,k)$ for $2 \le i,k \le 6$.

choosing the $D'$ in Theorem 3.1 so that each row of $R'$ had unit 1-norm, $\|(R')^{-1}\|_1$ never exceeded 111. One can also show that $O(\varepsilon)\|(R')^{-1}\|_1$ bounds the overall backward error $\frac{\|\delta X\|}{\|X\|} + \frac{\|\delta Y\|}{\|Y\|}$ in the computed SVD $U\Sigma V^T = (X+\delta X)D(Y+\delta Y)$, another way to confirm the accuracy.

As a further accuracy test we computed

$$\epsilon(i,k) = \max_{\kappa_2(X)=10^i, \kappa_2(Y)=10^k} \max_j \frac{|\sigma_{S,j} - \sigma_{D,j}|}{\sigma_{D,j}} \tag{17}$$

which should behave like $\varepsilon_S \cdot 10^{\max(i,k)} \approx 10^{\max(i,k)-7}$. This behavior is confirmed by the plot of $\epsilon(i,k)$ versus $i$ and $k$ in Figure 1.

# 4 Computing an Accurate RRD: Conditions on Scaled Condition Numbers

In this section we discuss conditions on the "scaled condition" of $G$ that permit an accurate rank revealing factorization (RRD) to be computed by straightforward GECP, by GE with other simple pivoting strategies, or by simply scaling the rows and/or columns of $G$. In other words, our conditions will depend only on $B$, where $G = D_1 B$, or $G = BD_2$ or $G = D_1 B D_2$, and $D_1$ and $D_2$ can be arbitrary diagonal matrices. We briefly review the simple case of one-sided scaling $G = BD_2$ with $B$ full column rank, and then discuss the general problem $G = D_1 B D_2$, which has a "combinatorial" analysis.

The simplest example, requiring nearly no computation at all, occurs when $B$ has full column rank, and we scale the columns $G = BD_2$. Then if $B$ is well-conditioned, the factorization $G = BD_2 \equiv XDY^T$ with $X = B$, $D = D_2$ and $Y = I$ is rank revealing. Thus we see that if $B$ is well-conditioned and of full-column rank, and we consider perturbations $G + \delta G = (B + \delta B)D_2$ where $\|\delta B\| \ll \|B\|$, then $G$'s SVD is determined to high relative accuracy independent of column scaling $D_2$. If we are given $G$ but not its factors $B$ and $D_2$, then we can recover nearly the best conditioned $B$ by simply dividing each column of $G$ by its 2-norm [53]. This discussion also applies to $G = D_1 B$, where $B$ is well-conditioned and has full row rank. Of course for this simple case it is unnecessary to compute an RRD in order to get an accurate SVD, but rather just apply one-sided Jacobi, a fact we exploited in step (3) of Algorithm 3.1, and which is further discussed in [19, 39, 21]. (The first paper remarking on the high accuracy of Jacobi appears to be [46].)

Extensions of these results to $G = D_1 B D_2$ (or $G = D_1 B$ with $B$ full column rank) are unavoidably combinatorial in nature, requiring conditions on *all* submatrices of $B$, not just $B$ itself. The first result of this kind for the SVD appears in [29], although related results for least squares problems appear in [50, 40, 55] (see section 4.2). In [29], Gu and Eisenstat show that for the $n$-by-$n$ matrices $G = D_1 B D_2$ and $G + \delta G = D_1(B + \delta B) D_2$ to have singular values agreeing to high relative accuracy, independent of $D_1$ and $D_2$, the smallest singular values of *all* square submatrices of $B$ must be large enough. In fact, they show that the relative error bound on the singular values is essentially $\|\delta B\|/\chi_1$, where $\chi_1$ is the smallest singular value of any square submatrix of $B$. There are exponentially many square submatrices, $\sum_{i=1}^{n} \binom{n}{i}^2 = O(4^n)$ of them. To see that this error bound is attainable, suppose without loss of generality that the submatrix $\hat{B}$ of $B$ with the smallest singular value is the leading $k$-by-$k$ submatrix of $B$. Then set the leading $k$ diagonal entries of $D_1$ and $D_2$ to one, and let the rest be very small. Then the $k$ largest singular values of $B$ are essentially the singular values of $\hat{B}$, and the $k$-th singular value is as sensitive as claimed. Note that even the *largest* singular value of $G$ may be the worst conditioned; suppose $0 < \epsilon \ll 1$ in the following example:

$$
\begin{aligned}
G &= \begin{bmatrix} 0 & 1 \\ 1 & \epsilon \end{bmatrix} = \begin{bmatrix} \frac{1}{\epsilon} & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ 0 & \epsilon \end{bmatrix} \\
G + \delta G &= \begin{bmatrix} 1 & 1 \\ 1 & \epsilon \end{bmatrix} = \begin{bmatrix} \frac{1}{\epsilon} & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \epsilon & 1 \\ 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ 0 & \epsilon \end{bmatrix} .
\end{aligned}
$$

Here, $G$ is nearly orthogonal, and $\|\delta G\| \approx \|G\|$. Thus, even a single zero matrix entry means that with appropriately chosen $D_1$ and $D_2$, the largest singular value can be ill-determined. Note that $D_1$, $D_2$ and $B$ are not uniquely determined by $G$; we could have chosen $D_1 = D_2 = I$ and $B = G$ in the above example, and concluded that all singular values of $G$ were well-conditioned. In practice $D_1$ and $D_2$ may be extra information supplied by the user along with $G$, in which case they determine the allowable uncertainty in entries of $G$. But we may also just be given $G$ without $D_1$ and $D_2$, in which case we would like an error bound corresponding to the "best" $D_1$ and $D_2$. These observations, and the exponential expense of computing $\chi_1$, motivates us to find a simpler, easily computable error bound.

Our error bound below will have the three attractive properties of

1. being small when $\chi_1$ is large,
2. costing just $O(n^3)$ to compute, and
3. being small just when $n$ particular submatrices of $B$ are well-conditioned, i.e. those determining the accuracy of $LU$ decomposition; the choice of submatrices will depend on $D_1$ and $D_2$.

To see the connection between well-conditioned submatrices of $B$ and accurate LU decomposition, recall that all intermediate results in LU decomposition can be expressed as Schur complements like $B_{22} - B_{21}B_{11}^{-1}B_{12}$ below:

$$B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} = \begin{bmatrix} I & 0 \\ B_{21}B_{11}^{-1} & I \end{bmatrix} \cdot \begin{bmatrix} B_{11} & B_{12} \\ 0 & B_{22} - B_{21}B_{11}^{-1}B_{12} \end{bmatrix}$$

Thus if all leading principal minors $B_{11}$ are well-conditioned, each Schur complement will be determined accurately. If we permit arbitrary diagonal scaling

$$G = D_1 B D_2 = \begin{bmatrix} D_{11} & 0 \\ 0 & D_{12} \end{bmatrix} \cdot \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} \cdot \begin{bmatrix} D_{21} & 0 \\ 0 & D_{22} \end{bmatrix}$$

then the diagonal matrices "factor through" the Schur complement in simple ways:

$$G = \begin{bmatrix} I & 0 \\ D_{12}(B_{12}B_{11}^{-1})D_{11}^{-1} & I \end{bmatrix} \cdot \begin{bmatrix} D_{11}B_{11}D_{21} & D_{11}B_{12}D_{22} \\ 0 & D_{12}(B_{22} - B_{21}B_{11}^{-1}B_{12})D_{22} \end{bmatrix} .$$

On the other hand, allowing arbitrary $D_1$ and $D_2$ means that *any* submatrix of $B$ is a candidate leading principal submatrix after GECP reorders rows and columns. This is why Gu and Eisenstat ask that *every* principal submatrix of $B$ be well-conditioned [29].

However, by assuming $D_1$ and $D_2$ are approximately sorted (with the diagonal entries more or less decreasing from top to bottom), which can always be achieved with row and/or column permutations of $G$, then we need only look at the conditioning of the $n$ leading principal submatrices of $B$, rather than all submatrices. This is because the natural pivot order is a good approximation of the one GECP would choose. This gives us a much cheaper criterion for high relative accuracy than Gu and Eisenstat, at the cost of a somewhat weaker bound.

In particular, suppose we have already permuted the rows and columns of $G$ so that $G = D_1 B D_2$ has the diagonal entries of $D_1$ and $D_2$ in roughly decreasing order. Define

$$\tau = \max_{1 \le i \le j \le n} \left\{ \frac{D_{1,j}}{D_{1,i}}, \frac{D_{2,j}}{D_{2,i}} \right\} . \tag{18}$$

Then $\tau$ is 1 exactly when the diagonal entries of $D_1$ and $D_2$ are in decreasing order, and $O(1)$ if they are more or less decreasing.

Let $B = LU$ be an LU decomposition of $B$ without pivoting, where $L$ and $U$ are lower and upper triangular, respectively, but not necessarily unit diagonal. The next theorem shows that if $L$ and $U$ are well-conditioned, which will be true if the smallest singular values of all leading principal submatrices of $B$ are large enough, and if $\tau = O(1)$, then the SVDs of $G$ and $G + \delta G = D_1(B + \delta B)D_2$ will agree to high relative accuracy when $\|\delta B\| / \|B\|$ is small.

**Theorem 4.1**

$$\begin{aligned} \frac{|\sigma_i(G+\delta G) - \sigma_i(G)|}{\max(\sigma_i(G+\delta G), \sigma_i(G))} &\le \frac{\tau\{\kappa(L) + \kappa(U)\} \cdot \max\{\|\bar{L} - I\|, \|\bar{U} - I\|\}}{1 - \tau\{\kappa(L) + \kappa(U)\} \cdot \max\{\|\bar{L} - I\|, \|\bar{U} - I\|\}} \\ &\le \tau\{\kappa(L) + \kappa(U)\} \cdot \|L^{-1}\| \cdot \|\delta B\| \cdot \|U^{-1}\| + O(\|\delta B\|^2) , \end{aligned}$$

*where* $I + L^{-1}\delta B U^{-1} = \bar{L}\bar{U}$ *is the LU decomposition without pivoting.*

*Proof:* We want to write $G+\delta G = D_L G D_R$, where $D_L$ and $D_R$ are close to identity matrices, and then apply Theorem 2.2. To this end, write

$$G+\delta G = D_1\left(LU+\delta B\right)D_2 = D_1 L\left(I+L^{-1}\delta B U^{-1}\right)UD_2 = D_1 L(\bar{L}\bar{U})UD_2 \; .$$

Hence

$$\begin{aligned} G+\delta G &= (D_1 L\bar{L})\cdot(I)\cdot(\bar{U}UD_2) \\ &= (D_1 L\bar{L})\cdot\left(L^{-1}D_1^{-1}GD_2^{-1}U^{-1}\right)\cdot(\bar{U}UD_2) \\ &= \left(D_1 L\bar{L}L^{-1}D_1^{-1}\right)\cdot G\cdot\left(D_2^{-1}U^{-1}\bar{U}UD_2\right) \\ &\equiv D_L\cdot G\cdot D_R. \end{aligned}$$

If $L$ and $U$ are well-conditioned, $L^{-1}\delta B U^{-1}$ will be small, so $\bar{L}$ and $\bar{U}$ are close to the identity. In fact, to first order in $\delta B$, we can write

$$\bar{L} = I + \mathrm{tril}\left(L^{-1}\delta B U^{-1}\right) \quad \text{and} \quad \bar{U} = I + \mathrm{triu}\left(L^{-1}\delta B U^{-1}\right), \tag{19}$$

where $\mathrm{tril}(X)$ is the strict lower triangular part of $X$, and $\mathrm{triu}(X)$ is the upper triangular part of $X$, including the diagonal (see also [52]). Thus

$$\|\bar{L}-I\| \le \|L^{-1}\|\cdot\|\delta B\|\cdot\|U^{-1}\| \quad \text{and} \quad \|\bar{U}-I\| \le \|L^{-1}\|\cdot\|\delta B\|\cdot\|U^{-1}\|,$$

and it follows that

$$\|D_L - I\| = \|D_1 L(\bar{L}-I)L^{-1}D_1^{-1}\| \le \tau\|L(\bar{L}-I)L^{-1}\| \le \tau\kappa(L)\cdot\|\bar{L}-I\|,$$

which is true in any absolute norm, since the $(i,j)$ entry of the lower triangular matrix $L(\bar{L}-I)L^{-1}$ is multiplied by $|D_{1,i}/D_{1,j}| \le \tau$. Similarly,

$$\|D_U - I\| \le \tau\kappa(U)\cdot\|\bar{U}-I\| \; .$$

Plugging these relations into Theorem 2.2 and simplifying, we obtain Theorem 4.1. □

One can also use Theorem 2.3 to prove a similar result about the singular vectors of $G$ and $G+\delta G$, but we will omit this.

We note that the higher order terms we dropped in Theorem 4.1 remain small as long as $\tau = O(1)$.

In general we will be given $G$, but not $D_1$ and $D_2$, so we can not sort them. Instead, we seek an a posteriori bound, that can be evaluated after GECP on $G$, that bounds the relative error in the SVD of $G$ introduced by roundoff during GECP; this bound will implicitly pick "good" $D_1$ and $D_2$. After stating this bound, we will relate it to the bound in Theorem 4.1.

**Theorem 4.2** *Let $P_1 G P_2 \approx \hat{L}\cdot\hat{U}$ be the factorization of $G$ computed by GECP in floating point arithmetic with machine precision $\varepsilon$. $\hat{L}$ is unit lower triangular, and $\hat{U}$ is upper triangular. To first order in $\varepsilon$, the relative error introduced in the singular values by roundoff during GECP is bounded by*

$$3n\varepsilon\{\|\,|\hat{L}|\cdot\mathrm{tril}(|\hat{L}^{-1}|\cdot|\hat{L}|\cdot|\hat{U}|\cdot|\hat{U}^{-1}|)\cdot|\hat{L}^{-1}|\,\| + \|\,|\hat{U}^{-1}|\cdot\mathrm{triu}(|\hat{L}^{-1}|\cdot|\hat{L}|\cdot|\hat{U}|\cdot|\hat{U}^{-1}|)\cdot|\hat{U}|\,\|\}.$$

*Proof:* Assume without loss of generality that the permutations involved in the factorization are identities. We use the fact that the backward error $\delta G$ in the decomposition $G + \delta G = \hat{L} \cdot \hat{U}$ is bounded component-wise by $|\delta G| \leq 3n\epsilon |\hat{L}| \cdot |\hat{U}|$. Then we factor $G = D_L^{-1} \left(G + \delta G\right) D_R^{-1}$ as follows:

$$G = \hat{L} \cdot \hat{U} - \delta G = \hat{L} \cdot \left(I - \hat{L}^{-1} \cdot \delta G \cdot \hat{U}^{-1}\right) \cdot \hat{U} = \hat{L} \cdot (\bar{L}\bar{U}) \cdot \hat{U},$$

where $I - \hat{L}^{-1}\delta G \hat{U}^{-1} = \bar{L}\bar{U}$ is the LU decomposition without pivoting. It follows that

$$\begin{aligned} G &= \left(\hat{L} \cdot \bar{L}\right) \cdot \left(\hat{L}^{-1} \left(G + \delta G\right) \hat{U}^{-1}\right) \cdot \left(\bar{U} \cdot \hat{U}\right) \\ &= \left(\hat{L} \cdot \bar{L} \cdot \hat{L}^{-1}\right) \cdot \left(G + \delta G\right) \cdot \left(\hat{U}^{-1} \cdot \bar{U} \cdot \hat{U}\right) \\ &\equiv D_L^{-1} \left(G + \delta G\right) D_R^{-1}. \end{aligned}$$

Taking absolute values, we see that

$$|D_L - I| = |\ \hat{L} \cdot \left(\bar{L}^{-1} - I\right) \cdot \hat{L}^{-1}\ |.$$

Similar to (19), we can write to first order that

$$|\ \bar{L}^{-1} - I\ | \leq 3n\epsilon \cdot \mathrm{tril}(\ |\hat{L}^{-1}| \cdot |\hat{L}| \cdot |\hat{U}| \cdot |\hat{U}^{-1}|\ ),$$

and hence

$$|D_L - I| \leq 3n\epsilon |\hat{L}| \cdot \mathrm{tril}(|\hat{L}^{-1}| \cdot |\hat{L}| \cdot |\hat{U}| \cdot |\hat{U}^{-1}|) \cdot |\hat{L}^{-1}|\ .$$

Similarly,

$$|D_R - I| \leq 3n\epsilon |\hat{U}^{-1}| \cdot \mathrm{triu}(|\hat{L}^{-1}| \cdot |\hat{L}| \cdot |\hat{U}| \cdot |\hat{U}^{-1}|) \cdot |\hat{U}|\ .$$

The theorem is proved by plugging these relations into Theorem 2.2 and dropping the higher order terms. □

Theorem 4.2 provides a computable error bound which can be used in practice. The diagonal matrices $D_1$ and $D_2$ do not appear explicitly in the expression. The expression in Theorem 4.2 does not change if we replace $\hat{L}$ and $\hat{U}$ by $\hat{L} \cdot D$ and $D^{-1} \cdot \hat{U}$, respectively, for any non-singular diagonal matrix $D$. Hence there is no need to choose $D_1$, $D_2$, and $D$ "optimally" to compute the error bound.

To better understand this bound, we now relate it to the bound in Theorem 4.1. We make the reasonable assumption that the permutations $P_1$ and $P_2$ computed by GECP nearly sort $D_1$ and $D_2$, so that $\tau$ in equation (18) is $O(1)$. This sorting property is a natural consequence of complete pivoting: it moves the largest potential pivot into the upper left corner. This lets us write $P_1 G P_2 \approx \hat{L}\hat{U} = D_1 L U D_2$, where we can take

$$\hat{L} = D_1 L D_1^{-1} \quad \text{and} \quad \hat{U} = D_1 U D_2\ .$$

Hence $L \cdot U$ is the LU decomposition of the unscaled matrix $D_1^{-1} \cdot (P_1 G P_2) \cdot D_2^{-1}$.

**Corollary 4.1** *To first order, the relative error introduced in the singular values by computing the decomposition $G = \hat{L} \cdot \hat{U}$ in floating point arithmetic with machine precision $\epsilon$ is bounded by*

$$3n\tau\epsilon \{\ \|\ |L| \cdot \mathrm{tril}(|L^{-1}| \cdot |L| \cdot |U| \cdot |U^{-1}|) \cdot |L^{-1}|\ \| + \|\ |U^{-1}| \cdot \mathrm{triu}(|L^{-1}| \cdot |L| \cdot |U| \cdot |U^{-1}|) \cdot |U|\ \|\ \},$$

*where $\tau$ is defined in (18). In other words, the relative error is small if $L$ and $U$ are well-conditioned, and $\tau = O(1)$.*

*Proof:* Similar to the proof of Theorem 4.1, it is easy to check that

$$\begin{aligned}\| \, |\hat{L}| \cdot \mathrm{tril}(|\hat{L}^{-1}| \cdot |\hat{L}| \cdot |\hat{U}| \cdot |\hat{U}^{-1}|) \cdot |\hat{L}^{-1}| \, \| &= \|D_1 \cdot |L| \cdot \mathrm{tril}(|L^{-1}| \cdot |L| \cdot |U| \cdot |U^{-1}|) \cdot |L^{-1}| \cdot D_1^{-1}\| \\ &\le \tau \cdot \| \, |L| \cdot \mathrm{tril}(|L^{-1}| \cdot |L| \cdot |U| \cdot |U^{-1}|) \cdot |L^{-1}| \, \| \, ,\end{aligned}$$

and

$$\| \, |\hat{U}^{-1}| \cdot \mathrm{triu}(|\hat{L}^{-1}| \cdot |\hat{L}| \cdot |\hat{U}| \cdot |\hat{U}^{-1}|) \cdot |\hat{U}| \, \| \le \tau \cdot \| \, |U^{-1}| \cdot \mathrm{triu}(|L^{-1}| \cdot |L| \cdot |U| \cdot |U^{-1}|) \cdot |U| \, \|.$$

Combining these relations with Theorem 4.2, Corollary 4.1 immediately follows. □

Similar to Theorem 4.2, the expression in Corollary 4.1 does not change if we replace $L$ and $U$ by $L \cdot D$ and $D^{-1} \cdot U$, respectively, for any non-singular diagonal matrix $D$. Hence there is no need to choose $D$ "optimally" to minimize the error bound.

As in Theorem 4.1, the higher order terms we dropped off in Theorem 4.2 and Corollary 4.1 remain higher order so long as $\tau = O(1)$.

The cases studied in [3] and [19], where the SVD is determined to high relative accuracy, are essentially subsumed by this analysis. In [3], *scaled diagonally dominant (s.d.d.)* matrices were studied, i.e. symmetric matrices of the form $G = D(E + N)D$, where $D$ was an arbitrary diagonal matrix, $E$ was diagonal with diagonal entries $\pm 1$, and $N$ satisfied $\|N\|_2 \equiv \gamma < 1$. It is easy to see that a symmetric permutation that sorts $D$ leaves all principle submatrices of $E + N$ with singular values between $1 - \gamma$ and $1 + \gamma$, i.e. well conditioned if $\gamma$ is small. In [19], symmetric positive definite matrices of the form $DAD$ were studied, where $D$ is any diagonal matrix and $A$ is symmetric positive definite and well-conditioned. Again, a symmetric permutation to sort $D$ leaves the leading principle submatrices of $A$ no worse conditioned than $A$ itself, by the Cauchy Interlace Theorem.

One may also ask how Corollary 4.1 is quantitatively related to the results of Gu and Eisenstat [29]. Unfortunately, the relation is not so obvious, other than that the expression is not large if the smallest singular values of all the principal submatrices of $B$ in $G = D_1 B D_2$ are large enough.

Given $G$, one can ask which $D_1$ and $D_2$ are "optimal", in the sense of giving a smallest error or perturbation bound. This can be reduced to linear programming, but since we see no use for it in practice, we will not present it here.

Finally, we briefly consider the case $G = D_1 B$, where $B$ has full column rank; the interesting case is when $B$ has more rows $m$ than columns $n$. In this case, we can do arbitrary column pivoting in the pursuit of a good LU decomposition, so that we expect a good decomposition (following Gu and Eisenstat) when *all* $k$-by-$n$ submatrices of $B$ have singular values that are not too small, or (following Corollary 4.1) when the $k$-by-$n$ submatrices of $B$ corresponding to the largest $n$ entries of $D_1$ are well-conditioned. We pursue this in section 4.2 below.

## 4.1 Numerical Experiments

In this section, we have four goals:

1. We want to assess the accuracy of GECP followed by Algorithm 3.1 in computing the SVD.

2. We want to assess the error bound in Theorem 4.2.

3. We want to show that GECP plays an essential role, by showing that QR with pivoting cannot be used in its place.

4. We want to show that the one-sided Jacobi algorithm from [19] is often as accurate as the more complicated algorithm proposed here (GECP plus Algorithm 3.1), but that it can fail.

We generate square random 200-by-200 matrices $G$ in the form $G = D_1 B D_2$, where $B$ is a random matrix with equilibrated column and row norms and with the spectral condition number $10^i$, $i = 1, 2, \ldots, 7$, and $D_1$ and $D_2$ are random diagonal matrices with $\kappa(D_2) = 10^k$, $k = 2, 4, 8, 10, 12, 14, 16$ and $\kappa(D_1) = \sqrt{\kappa(D_2)}$. For each fixed pair of parameters $(i, k)$ we use two different `MODE`s of singular value distributions for the triple $B$, $D_1$, $D_2$ (see section 3.1) where for each choice of the mode we generate 2 test matrices of the type $G = D_1 B D_2$. This makes a total of 196 test matrices, or 28 for each value of $i$.

Note that we do not directly control the condition numbers of the $LU$ factors of the (permuted) $B$ by this process, or the sorting of $D_1$ and $D_2$, which is what the error bounds ultimately depend on (see Theorem 4.1). Nonetheless, we argue that we roughly control $\|L^{-1}\|$ and $\|U^{-1}\|$, since $B = LU$ implies

$$\begin{aligned} \|B^{-1}\|^{1/2} &\le (\|U^{-1}\| \cdot \|L^{-1}\|)^{1/2} \\ &\le \max(\|U^{-1}\|, \|L^{-1}\|) \\ &= \max(\|B^{-1}L\|, \|UB^{-1}\|) \\ &\le \|B^{-1}\| \cdot \max(\|L\|, \|U\|) \ . \end{aligned}$$

Therefore, we choose the following error measure

$$\epsilon(i) = \max_{G = D_2 A D_1, \kappa(A) = 10^i} \max_k \frac{|\sigma_{S,k} - \sigma_{D,k}|}{\sigma_{D,k}} \ . \tag{20}$$

For GECP followed by Algorithm 3.1 we expect $\epsilon(i) \lessapprox \varepsilon_S 10^i \approx 10^{7-i}$, for the reasons just discussed. Table 2 displays the computed results:

- Row 1, Column 2: The expected relative error $10^{i-7}$ for GECP and Algorithm 3.1 applied to the 28 test matrices with $\kappa(A) = 10^i$.
- Row 1, Column 3: The maximum measured relative error $\epsilon(i)$ for GECP and Algorithm 3.1 on the same set of matrices.
- Row 1, Column 4: The bound from Theorem 4.2 for $\epsilon(i)$ for GECP and Algorithm 3.1 for the same set of matrices *computed in double precision.*
- Row 2, Column 2: The maximum measured relative error $\epsilon(i)$ for the algorithm consisting of QR decomposition followed by one-sided Jacobi applied to R for the same set of matrices.
- Row 2, Column 3: The maximum measured relative error $\epsilon(i)$ for the one-sided Jacobi algorithm alone for the same set of matrices.

Since columns 2 and 3 in row 1 of Table 2 roughly agree, GECP followed by Algorithm 3.1 is as accurate as predicted. Column 4 of row 1 shows that the error bound from Theorem 4.2 predicts that the *double precision* algorithm delivers at least about 5 to 8 digits of accuracy, which is pessimistic, but useful. The corresponding error bounds for the *single precision* algorithm are all about $\varepsilon_S/\varepsilon_D \approx 5 \cdot 10^8$ times larger, and so all $O(1)$ or larger. In other words, the bound of

| $i$ | Expected $\epsilon(i)$ for GECP + Algorithm 3.1 in single precision | Measured $\epsilon(i)$ for GECP + Algorithm 3.1 in single precision | Bound for $\epsilon(i)$ for GECP + Algorithm 3.1 from Theorem 4.2 in double precision |
|---|---|---|---|
| 1 | $10^{-6}$ | $4 \cdot 10^{-6}$ | $2 \cdot 10^{-8}$ |
| 2 | $10^{-5}$ | $9 \cdot 10^{-6}$ | $3 \cdot 10^{-8}$ |
| 3 | $10^{-4}$ | $1 \cdot 10^{-4}$ | $3 \cdot 10^{-8}$ |
| 4 | $10^{-3}$ | $1 \cdot 10^{-3}$ | $4 \cdot 10^{-8}$ |
| 5 | $10^{-2}$ | $8 \cdot 10^{-3}$ | $1 \cdot 10^{-7}$ |
| 6 | $10^{-1}$ | $1 \cdot 10^{-1}$ | $1 \cdot 10^{-6}$ |
| 7 | 1 | 1 | $1 \cdot 10^{-5}$ |

| $i$ | Measured $\epsilon(i)$ for QR + Jacobi SVD in single precision | Measured $\epsilon(i)$ for Jacobi SVD in single precision |
|---|---|---|
| 1 | 0.7 | $5 \cdot 10^{-6}$ |
| 2 | 0.9 | $8 \cdot 10^{-6}$ |
| 3 | $> 2$ | $1 \cdot 10^{-4}$ |
| 4 | $> 50$ | $9 \cdot 10^{-4}$ |
| 5 | $> 300$ | $7 \cdot 10^{-3}$ |
| 6 | $> 1000$ | 0.2 |
| 7 | $> 1000$ | 1 |

Table 2: Accuracy of Overall SVD Algorithms

Theorem 4.2 provides useful bounds in double precision, but is too pessimistic to be useful in single precision.

Column 2 of row 2 shows that QR combined with one-sided Jacobi is *not* nearly as accurate an SVD algorithm. Existing theory [19] guarantees high relative accuracy for the algorithms in columns 2 and 3 of row 2 only when $G$ is scaled from one side ($G = BD_1$). Therefore, it is something of a surprise that Column 3 of row 2 shows one-sided Jacobi to be about as accurate as our more sophisticated algorithm GECP with Algorithm 3.1. This leads us to ask whether there are examples where GECP with Algorithm 3.1 is significantly more accurate than simple one-sided Jacobi.

The following 3-by-3 example shows how one-sided Jacobi can fail when the new algorithm succeeds. Let $\gamma$ and $\delta$ satisfy $1 \gg \gamma \gg \delta > 0$; $\gamma = 10^{-20}$ and $\delta = 10^{-40}$ will do in double precision. Let

$$\begin{aligned} G &= \begin{bmatrix} 1 & \gamma & \gamma \\ -\gamma & \gamma & \gamma^2 \\ 0 & \delta & 0 \end{bmatrix} \\ &= D_1 \cdot B \cdot D_2 \\ &\equiv \begin{bmatrix} 1 & & \\ & \gamma & \\ & & \delta \end{bmatrix} \cdot \begin{bmatrix} 1 & \gamma & 1 \\ -1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 & & \\ & 1 & \\ & & \gamma \end{bmatrix} . \end{aligned}$$

Since $D_1$ and $D_2$ are sorted, and the leading principal minors of $B$ are well-conditioned, the SVD of $G$ is determined to high relative accuracy. GECP applied to $G$ requires no pivot exchanges, and yields very accurate $LDU$ factors, with $L$ and $U$ nearly identity matrices, so that Algorithm 3.1 computes a very accurate SVD with singular values nearly equal to 1, $\gamma$, and $2\gamma\delta$.

But if we apply one-sided Jacobi to the right of $G$ so that it rotates columns 2 and 3 first, then we lose all accuracy in the smallest singular value $2\gamma\delta$. (In a Matlab experiment with $\gamma = 10^{-20}$ and $\delta = 10^{-40}$, we get $5 \cdot 10^{-57}$ instead of $2 \cdot 10^{-60}$.) This occurs because the first Jacobi rotation angle is $O(1)$ rather than $O(\gamma)$, which does not respect the column scaling, and so causes a large backward error in $B$.

### 4.2 Relationship to Weighted Least Squares Problems

In [50, 40, 55, 54, 34], the weighted least squares problem $\min_x \|D^{1/2}(Ax - b)\|_2$ is considered, with the goal of deriving algorithms and error bounds that hold independent of the diagonal scaling matrix $D$. In these papers it is shown that the condition number essentially depends on the following combinatorial object: Suppose $A$ is full rank and $m$-by-$n$, and let $x_{WLS}$ be the unique solution of the weighted least squares problem. Let $Q$ be any $m$-by-$n$ unitary matrix with the same column space as $A$; the $Q$ from the QR decomposition of $A$ will do. Let $\rho$ be the smallest nonzero singular value of any $k$-by-$n$ submatrix of $Q$, for $1 \le k \le m$. Then the norm of the "weighted projector" $P_D \equiv A(A^TDA)^{-1}A^TD$ that maps $b$ to its best approximation $Ax_{WLS} = P_Db$ is at most $\|P_D\| \le \rho^{-1}$, independent of $D$. It is easy to confirm that $P_D$ does not change if $A$ is postmultiplied by any nonsingular $n$-by-$n$ matrix, which is why we can replace $A$ by $Q$. To see why the combinatorial definition of $\rho$ is natural, we can let $D$ have $k$ diagonal entries equal to 1 and the rest very small; this essentially selects a $k$-by-$n$ submatrix $\hat{Q}$ of $Q$, with the large columns of $P_D$ approximately given by $Q(\hat{Q}^T\hat{Q})^{-1}\hat{Q}$, whose norm is the reciprocal of smallest singular value of $\hat{Q}$.

Since we are interested in the whole SVD, not just the pseudoinverse, we cannot replace $A$ by $Q$. But suppose that $A$ were unitary (or just well-conditioned). Then the conditions imposed in [50, 40, 55, 34] are essentially the same as the conditions in the last paragraph of the first part of section 4.

# 5 Computing an Accurate RRD: Combinatorial and Algebraic Conditions

In this section we will discuss *combinatorial and algebraic conditions* on $G$ i.e. conditions on $G$'s sparsity pattern and sign pattern, or on algebraic relationships among the entries of $G$, that guarantee that we can perform Gaussian elimination with pivoting to get an accurate RRD of $G$. Then we will use Algorithm 3.1 to compute the SVD of $G$. In this section we just motivate and outline these conditions, and leave the details to subsequent sections.

We begin with the fact that every final or intermediate value computed by Gaussian elimination, with any legal pivot order (i.e. not leading to divide-by-zero), is either *a minor or quotient of minors of* $G$. The conditions we impose on $G$ will guarantee that *all* minors of $G$ can be computed accurately. Since the quotient of two values known to high relative accuracy is also known to high relative accuracy, this implies that $L$, $D$ and $U$ can be computed accurately, for any legal pivot order.

More specifically, we will use the following classical result:

**Lemma 5.1** Let $G = P_r LDUP_c$ be any factorization of $G$ computed by Gaussian elimination with any pivot order not leading to division by zero. Here $P_r$ and $P_c$ are permutations, $L$ and $U$ are unit triangular, and $D$ is diagonal. We also write this as $G = P_r \bar{L}\bar{U}P_c$, where either $\bar{L} = L$ and $\bar{U} = DU$, or $\bar{U} = U$ and $\bar{L} = LD$. Let $G_s$ be any Schur complement of $G$, i.e.

$$G_s = G_{22} - G_{21}G_{11}^{-1}G_{12} \quad \text{where} \quad G = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} .$$

Then

1. Every minor of $G_s$ is a quotient of minors (or just a minor) of $G$.
2. Every minor of $G^{-1}$ is plus or minus a quotient of minors of $G$.
3. Every entry of $L$, $D$, $U$, $\bar{L}$ and $\bar{U}$ is either zero or a quotient of minors (or just a minor) of $G$.
4. Every minor of $L$, $D$, $U$, $\bar{L}$ and $\bar{U}$ consisting of consecutive rows is either zero or a quotient of minors (or just a minor) of $G$.
5. Every entry of $L^{-1}$, $D^{-1}$, $U^{-1}$, $\bar{L}^{-1}$ and $\bar{U}^{-1}$ is either zero or a quotient of minors (or just a minor) of $G$.

These facts follow from Sylvester's determinant identity [35], or more specifically from observations like

$$D_{n,n} = \frac{D_{1,1} \cdots D_{n-1,n-1} \cdot D_{n,n}}{D_{1,1} \cdots D_{n-1,n-1}} = \frac{\det(G)}{\det(G(1:n-1, 1:n-1))}$$

(assuming $P_r = P_c = I$).

To see what conditions we must impose on $G$ to compute its minors accurately, let us consider a general algebraic expression $e(\zeta_1, \zeta_2, ..., \zeta_k)$, where $e(\cdot)$ is defined by a fixed sequence of additions, subtractions, multiplications and divisions. We assume that each real datum $\zeta_i$ is known to high relative accuracy, and we may also know its sign. For $e$ to be determined to high relative accuracy by the data $\zeta_i$, *independent of the magnitudes of the* $\zeta_i$, it is clearly necessary and sufficient for $e$ to be defined by

**(1)** multiplications and divisions, and

**(2)** addition of quantities with the same sign.

For example, $(\zeta_1^2 + \zeta_2^2) \cdot \zeta_3^3 / \zeta_4$ is accurately determined, but $\zeta_1 + \zeta_2$ is not unless it is also known that $\zeta_1 \zeta_2 > 0$. In other words, the only forbidden operation is true subtraction, because cancellation in leading digits can leave the sum $s = \zeta_1 + \zeta_2$ with arbitrarily less relative accuracy than in $\zeta_1$ or in $\zeta_2$, if $\zeta_1$ and $-\zeta_2$ are close.

Here is a more formal way to describe this property of $e$: Let $\hat{e}(\cdot)$ be another expression which differs from $e(\cdot)$ only by having the result of each operation multiplied by a different $1 + \delta$, where $|\delta| \le \varepsilon \ll 1$; in other words $\hat{e}()$ is the value of $e$ computed in floating point with machine precision $\varepsilon$. (Here and elsewhere we ignore the possibility of over/underflow.) Also let $|\hat{\zeta}_i - \zeta_i| \le \eta |\zeta_i|$ for all $i$, where $0 < \eta \ll 1$. Then

$$\frac{|\hat{e}(\hat{\zeta}_1, ..., \hat{\zeta}_k) - e(\zeta_1, ..., \zeta_k)|}{|e(\zeta_1, ..., \zeta_k)|} = O(\max(\eta, \varepsilon)) \quad . \tag{21}$$

Requirements (1) and (2) leave us enough freedom to find accurate expressions $e$ for the minors of a variety of interesting matrix classes, as we outline below. But we can solve still more problems if we permit ourselves one more operation, which is only justified in (well-implemented) floating point arithmetic:
**(3)** addition or subtraction if the operands are original data, i.e. $\zeta_i$ and $\zeta_j$.

Here is the justification. Consider an expression $e(\zeta_1, ..., \zeta_k)$ consisting of operations (1), (2) and (3), and where the $\zeta_i$ are floating point numbers, and let $\hat{e}(\cdot)$ be the floating point version of $e$ as above. Then

$$\frac{|\hat{e}(\zeta_1, ..., \zeta_k) - e(\zeta_1, ..., \zeta_k)|}{|e(\zeta_1, ..., \zeta_k)|} = O(\varepsilon) \quad . \tag{22}$$

Equation (22) holds because if there is cancellation in computing the sum or difference $\zeta_i - \zeta_j$ of two floating point numbers, then the sum or difference is *exact* [4]. Comparing with equation (21), we see that we compute an approximation $\hat{e}(\zeta_i)$ of the true value $e(\zeta_i)$ to high relative accuracy, but if the $\zeta_i$ themselves are uncertain, there is no bound on the difference $\hat{e}(\hat{\zeta}_i) - e(\zeta_i)$. Put another way, an algorithm based on property (3) will compute accurate minors (and eventually an accurate SVD) of the problem *as stored in the machine*, even if the minors and SVD of the problem in the machine are very sensitive to changes in $\zeta_i$. But if we do not need to use (3), then we have the further information that small relative changes in the $\zeta_i$ would not have significantly changed the minors or SVD.

[4]This assumes addition and subtraction are implemented with guard digits, and so excludes the Cray T90 and its predecessors and emulators, like the NEC SX machines.

Armed with this understanding of which expressions $e$ can be evaluated accurately, consider just the determinant of $G$ itself, which we assume to be $n$-by-$n$. Its Laplace expansion is

$$\det(G) = \sum_p [\mathrm{sign}(p) \cdot g_{1,p_1} \cdot g_{2,p_2} \cdots g_{n,p_n}] \tag{23}$$

where the sum is over all permutations $p = (p_1, ..., p_n)$ of $(1, ..., n)$, and $\mathrm{sign}(p) = \pm 1$ is the sign of the permutation $p$. This is a sum and difference of monomials $\prod_i g_{i,p_i}$. We want to know when an expression $e = \det(G)$ exists satisfying requirements (1), (2) and possibly (3). We begin by assuming that the entries $g_{i,j}$ are themselves the initial data ($\zeta_i$s), and each nonzero $g_{i,j}$ is only known to high relative accuracy. Later we will consider the case when the $g_{i,j}$ are given as algebraic expression in the initial data.

Think of $G$ as having a fixed sparsity pattern, so some $g_{i,j}$ are known to be zero. Any monomials containing such a zero factor are identically zero. Any monomial that is a product of $n$ nonzero terms is determined to high relative accuracy. If we further fix the signs of each $g_{i,j}$, then each nonzero monomial will have a fixed sign as well.

So when is the expansion (23) of $\det(G)$ determined to high relative accuracy by the initial data $g_{i,j}$? There are 3 cases:

1. If all monomials are exactly 0, because each contains a zero $g_{i,j}$, then $\det(G)$ is exactly 0 (to high relative accuracy!).

2. If exactly one monomial is nonzero, then $\det(G)$ is determined to high relative accuracy, since the monomial is.

3. If two or more monomials are nonzero, and the $g_{i,j}$ have independent small relative errors and independent signs, then cancellation can destroy relative accuracy in the sum. We can avoid cancellation and guarantee high relative accuracy if the signs of the $g_{i,j}$ are restricted so that all nonzero monomials have the same sign.

We claim that those matrices, *all* of whose minors have 0 or 1 nonzero monomials in their Laplace expansions, are precisely the matrices whose graphs are acyclic [17]; we define this further in section 6 below. In other words, there is a simple necessary and sufficient condition for a sparse matrix to have each minor either zero or equal to a single monomial, and so determined to high relative accuracy. For these matrices, which have been extensively studied before, there are several available high accuracy SVD algorithms.

If all the minors of a matrix have Laplace expansions where each nonzero monomial has the same sign, as described in case 3 above, then the matrix is called *total signed compound (TSC)* as defined in [11]. We discuss this in detail in section 7 below. In other words, we can completely characterize which sparsity patterns (the acyclic ones), or which sparsity and sign patterns (the TSC ones) guarantee accurate minors, and so accurate LDU factors and an accurate SVD.

In both these cases, acyclic matrices and TSC matrices, straightforward GECP will *not* determine the entries of $L$, $D$ and $U$ to high relative accuracy. This is because cancellation may occur. In other words, we need to modify GECP, based on the special structure of these matrices, that compute the same LDU factorization but without cancellation. We discuss these algorithms below. Unfortunately, their complexities can be larger than for GECP ($O(n^4)$ instead of $O(n^3)$ for TSC matrices); it is an open problem to find faster algorithms.

Now we consider the case where the matrix entries $g_{i,j}$ are not the initial data, but rather algebraic expressions in the initial data. In subsequent sections we discuss 3 examples in detail:

1. Suppose $G = D_L Z D_R$, where $D_L$ and $D_R$ are diagonal matrices with the initial data on their diagonals, and $Z$ is any fixed matrix. Then $\det(G) = \det(D_L) \cdot \det(Z) \cdot \det(D_R)$ is the product of initial data (the diagonal entries of $D_L$ and $D_R$) and a fixed constant ($\det(Z)$), and so determined to high relative accuracy. Clearly, the same is true of any minor of $G$. An important special case occurs when $Z$ is an integer matrix with each minor equal to $-1$, 0, or $+1$. Such a $Z$ is called *totally unimodular (TU)* [10], and we correspondingly call $G$ *diagonally scaled totally unimodular (DSTU)*. DSTU matrices include both acyclic matrices and finite element matrices arising from linear mass-spring systems as special cases; these matrices are discussed in sections 8 and 12.1, respectively. It is particularly easy to modify Gaussian elimination to attain high relative accuracy on DSTU matrices.

2. *Cauchy matrices* are matrices of the form $g_{i,j} = 1/(x_i - y_j)$, where the $x_i$ and $y_j$ are initial data. There is a classical formula for any minor of a Cauchy matrix that satisfies requirements (1), (2) and (3). The modification of GECP required to attain high relative accuracy appears to cost $O(n^5)$. This is discussed in section 9.

3. *Totally positive matrices* are matrices each of whose minors is positive. They arise in many situations in applied mathematics [35]. It turns out that one can construct all totally positive matrices from simpler ones by repeatedly using a set of *composition laws*. These laws also turn out to provide high accuracy formulas for all minors in terms of high accuracy formulas for simpler minors. Many of these formulas turn out to be exponentially expensive, and it remains an open problem to find efficient (or just polynomially expensive!) formulas for all minors. This is discussed in section 10.

# 6 Acyclic Matrices

Some of this material originally appeared in [17]; we summarize it here for completeness. Let $\mathcal{G}$ be the class of matrices with a given sparsity pattern, i.e. the locations of the nonzero entries are given. We let $\mathrm{Gr}(\mathcal{G})$ denote the *graph of* $\mathcal{G}$, i.e. the bipartite graph with one node for each row, one node for each column, and an edge $(i,j)$ if and only if entry $(i,j)$ is allowed to be nonzero.

**Theorem 6.1** *The following three conditions are* equivalent*:*

1. $\mathrm{Gr}(\mathcal{G})$ *is acyclic.*

2. *For all matrices $G \in \mathcal{G}$, and for any pivot sequence that does not divide by zero, small relative changes in the entries of $G$ cause only small relative changes in the entries of $L$, $D$ and $U$ computed by Gaussian elimination.*

3. *For all matrices $G \in \mathcal{G}$, small relative changes in $G$ cause only small relative perturbations in the SVD, in the sense of bounds (7) and (8).*

Acyclic matrices include bidiagonal matrices, "broken arrow" matrices (which are nonzero only on the diagonal and in one row or one column), and exponentially many other permutation-inequivalent patterns. All acyclic matrices are very sparse, with at most $2n-1$ nonzeros in an $n$-by-$n$ acyclic matrix.

We sketch the proof of Theorem 6.1; details are in [17]. Let $\mathrm{Gr}(G)$ be defined for a particular matrix $G$ just as is was defined for a class $\mathcal{G}$ above: there is one node per row, one node per column, and an edge $(i,j)$ if and only if $g_{i,j} \neq 0$. Recall that a *perfect matching* in a graph with $2n$ nodes is a set of $n$ edges where each node is the endpoint of exactly one edge. We depend on the elementary fact that there is a one-to-one correspondence between the monomials in the determinant expansion of any matrix $G$, and perfect matchings in $\mathrm{Gr}(G)$: Each monomial corresponds to a unique choice of $n$ nonzeros in $G$, one in each row and one in each column; each such set of $n$ nonzeros corresponds to $n$ edges forming a perfect matching between the $n$ row nodes and $n$ columns nodes. It is a simple graph theoretic lemma that a bipartite graph is acyclic if and only if each subgraph has at most one perfect matching (a cycle can be used to construct two perfect matchings, and vice versa). In other words $\mathrm{Gr}(G)$ is acyclic if the determinant expansion of each submatrix of $G$ has at most one nonzero term, which is equivalent to each minor being determined to high relative accuracy, which is sufficient for an accurate LDU decomposition, and an accurate SVD. To see that $\mathrm{Gr}(\mathcal{G})$ being acyclic is necessary, note that if there are two or more terms in the determinant expansion of some $k$-by-$k$ minor, then we can choose the matrix entries so that the minor is zero because of cancellation, and the matrix outside the $k$-by-$k$ submatrix defining the minor is exactly zero. Then both $D_{kk}$ and $\sigma_k$ are exactly zero, but become nonzero with arbitrarily small perturbations of any matrix entry that makes the minor nonzero. In other words, if the graph is cyclic, neither the SVD nor LU decomposition may be determined to high relative accuracy, for certain values of the matrix entries.

See Theorem 8.2 below for quantitative bounds on the accuracy with which the SVD is determined.

We can sometimes take advantage of the acyclic structure to compute the SVD quickly. For example, if the matrix is bidiagonal, various algorithms based on QR [18, 16] and QD [25] are available. For singular values of general acyclic matrices, bisection [17] is available, but until now no relatively accurate algorithm for the singular vectors was available.

We defer discussion of the algorithm for high accuracy LDU factorization of an acyclic matrix to section 8, where we present it as a special case of a more general algorithm.

# 7 Total Signed Compound (TSC) Matrices

The following definitions are taken from [11]. Let $\mathcal{S}$ be the set of all matrices with a given sparsity and sign pattern, i.e. the locations and signs of the nonzero entries are given. For example $\mathcal{S}$ could be the set of all square matrices with positive numbers on the main diagonal, negative numbers on the first superdiagonal, and zeros elsewhere. $\mathcal{S}$ is called *sign nonsingular (SNS)* if it contains only square matrices, and the Laplace expansion (23) of the determinant of each $G \in \mathcal{S}$ is the sum of monomials of like-sign, with at least one nonzero monomial. $\mathcal{S}$ is called *total signed compound (TSC)* if every square submatrix of any $G \in \mathcal{S}$ is either SNS, or structurally singular (i.e. no nonzero monomials appear in its determinant expansion).

Another, constructive definition of TSC matrices is as follows [11, 47]. We will need it later for our algorithm. Every TSC matrix can be obtained by starting with a 1-by-1 nonzero matrix and applying the following four construction rules repeatedly in some order:

1. If $G$ is TSC then permuting the rows, permuting the columns, or multiplying a row or column by $-1$ leaves $G$ TSC.

2. If $G_1$ and $G_2$ are TSC, so is the direct sum $G = \begin{bmatrix} G_1 & 0 \\ 0 & G_2 \end{bmatrix}$.

3. If $G_1 = \begin{bmatrix} G_1' \\ x_1^T \end{bmatrix}$ and $G_2 = \begin{bmatrix} x_2^T \\ G_2' \end{bmatrix}$ are TSC, where $x_1^T$ is the last row of $G_1$ and $x_2^T$ is the first row of $G_2$, then then so is the *weak direct sum* $G = \begin{bmatrix} G_1' & 0 \\ x_1^T & x_2^T \\ 0 & G_2' \end{bmatrix}$. Weak direct sums can also be formed by having $G_1$ and $G_2$ overlap in one column.

4. If the $m$-by-$n$ matrix $G'$ is TSC, with $G'_{ij} \neq 0$, then so is the $m+1$-by-$n+1$ matrix $G$ obtained as follows:

$$G = \left[ \begin{array}{c|c} & 0 \\ & \vdots \\ & 0 \\ G' & G_{i,n+1} \\ & 0 \\ & \vdots \\ & 0 \\ \hline 0, \cdots 0, G_{m+1,j}, 0, \cdots 0 & G_{m+1,n+1} \end{array} \right] \tag{24}$$

where we can also set $G'_{i,j}$ to zero. The new possibly nonzero entries $G_{m+1,n+1}$, $G_{m+1,j}$ and $G_{i,n+1}$ must be chosen so that the two monomials in the minor $G_{m+1,n+1} \cdot G'_{i,j} - G_{i,n+1} \cdot G_{m+1,j}$ have the same sign (or are zero).

It is easy to confirm that the examples in (11) are obtained by repeated application of construction (4) above.

TSC matrices are quite sparse, as the following lemma shows:

**Lemma 7.1** An $m$-by-$n$ TSC matrix has at most $1.5(m+n)-2$ nonzero entries.

*Proof:* We use induction on $m+n$, and the fact that a large TSC matrix is built from smaller TSC matrices according to construction rules 2, 3 and 4 (rule 1 does not change the nonzero count). The formula is obviously true when $m = n = 1$. In rule 2, suppose $G_i$ is $m_i$-by-$n_i$. By induction $G_i$ has at most $1.5(m_i+n_i)-2$ nonzeros, and it is easy to confirm that $G$ has at most $[1.5(m_1+n_1)-2]+[1.5(m_2+n_2)-2] = 1.5(m+n)-4 < 1.5(m+n)-2$ nonzeros, as desired. Rule 3 is similar. For rule 4 we use the fact that $G$ has at most 3 more nonzeros than $G'$. $\Box$

**Theorem 7.1** *The following three conditions are* equivalent*:*

1. *$\mathcal{S}$ is TSC.*

2. *For all matrices $G \in \mathcal{S}$, and for any pivot sequence that does not divide by zero, small relative changes in the entries of $G$ cause only small relative changes in the entries of $L$, $D$ and $U$ computed by Gaussian elimination.*

3. *For all matrices $G \in \mathcal{S}$, small relative changes in $G$ cause only small relative perturbations in the SVD, in the sense of bounds (7) and (8).*

*Proof:* The proof is entirely analogous to that of Theorem 6.1. We use the lack of cancellation in all minors to conclude that all entries of $L$, $D$ and $U$ are determined to high relative accuracy, and hence that the SVD is determined to high accuracy. Similarly, If $G$ is not TSC, we can construct a matrix where a minor vanishes by cancellation, so that some entry of $D$, or some singular value, is "accidentally" zero. $\Box$

To show how to modify the standard Gaussian elimination algorithm to factor TSC matrices with high relative accuracy, we need the following lemma:

**Lemma 7.2** There is an algorithm for computing the determinant of an $n$-by-$n$ TSC matrix to high relative accuracy, that requires at most $4n-1$ floating point operations.

*Proof:* As in the last lemma, we will use induction on $n$, exploiting the constructibility of any TSC matrix using the 4 rules above. A practical algorithm would represent a TSC matrix as a tree whose nodes represent applications of the 4 rules, processing the tree in topological order, but we will omit these details. In particular, we will not discuss the complexity of building this tree (which could possibly exceed the cost of the floating point operations).

Rule 1 has a trivial effect on the determinant, either leaving it unchanged or negating it.

If $G$ is constructed by Rule 2, then there are two cases, depending on whether $G_1$ and $G_2$ are both square or not. If they are not square, $\det(G)$ is clearly 0. Otherwise, $\det(G) = \det(G_1)\det(G_2)$, which computes $\det(G)$ to high relative accuracy from $\det(G_1)$ and $\det(G_2)$, at the cost of 1 flop plus the costs of $\det(G_1)$ and $\det(G_2)$.

Now suppose $G$ is constructed by Rule 3, where $G_i$ is $m_i$-by-$n_i$. Since $G$ is square, $n_1 + n_2 = m_1 + m_2 - 1 \equiv n$. For $\det(G)$ to be nonzero, the two zero blocks in $G$ can not be too big; in particular we need $m_1 - 1 + n_2 \le n$ and $m_2 - 1 + n_1 \le n$. There are only two solutions of these simultaneous inequalities and equations: $m_1 = n_1$ and $m_2 - 1 = n_2$, or $m_1 - 1 = n_1$ and $m_2 = n_2$. In the first case $\det(G) = \det(G_1)\det(G'_2)$, and in the second case $\det(G) = \det(G'_1)\det(G_2)$, either of which costs 1 flop plus the costs of $\det(G_1)$ and $\det(G'_2)$ (or $\det(G'_1)$ and $\det(G_2)$).

Finally, consider Rule 4. If $G_{m+1,n+1} \neq 0$, then we can do one step of Gaussian elimination starting from the bottom of the matrix in equation (24) to get

$$G = \left[\begin{array}{c|c} & 0 \\ & \vdots \\ & 0 \\ I & G_{i,n+1}/G_{m+1,n+1} \\ & 0 \\ & \vdots \\ & 0 \\ \hline 0 & 1 \end{array}\right] \cdot \left[\begin{array}{c|c} G'' & 0 \\ \hline 0, \cdots 0, G_{m+1,j}, 0, \cdots 0 & G_{m+1,n+1} \end{array}\right]$$

where $G'' = G'$ except that $G'_{i,j}$ has been changed to $G''_{i,j} = G'_{i,j} - G'_{i,n+1}G'_{m+1,j}/G'_{m+1,n+1}$; there is no cancellation in this formula because of the TSC property, so $G''_{i,j}$ is computed to high relative accuracy, and $G''$ is still TSC. Then $\det(G) = G_{m+1,n+1} \cdot \det(G'')$, which costs 4 flops plus the cost of $\det(G'')$. If $G_{m+1,n+1} = 0$, we expand by minors in the last column, so $\det(G) = \pm G_{i,n+1} \cdot G_{m+1,j} \cdot \det(G''')$, where $G'''$ is the $n-2$-by-$n-2$ submatrix of $G'$ lying outside row $i$ and column $j$. This costs 2 flops plus the cost of $\det(G''')$.

If $C_n$ is the maximum cost of $\det(G)$ when $G$ is $n$-by-$n$, then combining the above 4 rules yields

$$C_n \le \max( \max_{n_1+n_2=n} (1 + C_{n_1} + C_{n_2}), 3 + C_{n-1}, 2 + C_{n-2})$$

which has solution $C_n = 4n - 1$. $\Box$

Here is a simple version of Gaussian elimination with pivoting for TSC matrices, that computes all entries of $L$, $D$ and $U$ to high relative accuracy:

**Algorithm 7.1** Performing Gaussian elimination with pivoting $G = P_r LDUP_c$ in a forward stable manner on an $m$-by-$n$ TSC matrix $G$, where $m \ge n$.

```
        for i = 1 to min(m − 1, n)
            pivot so that G_ii is nonzero
            D_ii = G_ii
            for j = i + 1 to m
                L_ji = G_ji / D_ii
            endfor
            for j = i + 1 to m
                for k = i + 1 to n
(*)                 G_jk = G_jk − L_ji · G_ik
(**)                If the last subtraction has two nonzero operands with the same sign, then
                        recompute G_jk as the quotient of two minors, each computed using Lemma 7.2
                endfor
            endfor
            for k = i + 1 to n
                U_ik = G_ik / D_ii
            endfor
```

The above algorithm is essentially identical to conventional Gaussian elimination, except for line (**).

**Theorem 7.2** *Algorithm 7.1 computes all the entries of the $L$, $D$ and $U$ factors of a TSC matrix $G$ to high relative accuracy, for any pivot sequence not dividing by zero. The cost of the algorithm is $O(mn^3)$.*

*Proof:* The only possible source of forward instability in Algorithm 7.1 is the subtraction in line (*), and line (**) is a "brute force" way to recompute the result of line (*) so as to guarantee high relative accuracy. The complexity bound follows since line (**) can cost as much as $O(n)$ by Lemma 7.2. $\Box$

If the pivot sequence is given in advance, the complexity drops from $O(n^4)$ to $O(n^3)$, because we do not need to compute all entries of all Schur complements in order to choose the maximum pivot at each step. For an example illustrating how the complexity can be as large as $O(n^4)$, consider

$$G = \begin{bmatrix} 1 & y^T \\ x & DP \end{bmatrix}$$

where $x$ and $y$ have positive entries just less than 1, $D$ is a negative definite diagonal matrix with entries of tiny but widely varying magnitudes, and $P$ is a permutation matrix. In this case $L$ and

$U$ can be dense, with Lemma 7.2 *always* invoked in line (**). Note that we need to recompute the entries of intermediate Schur complements accurately enough to choose the correct pivot sequence. One could probably modify the test in line (**) to invoke Lemma 7.2 less frequently, for example, only when cancellation is severe enough.

We have not developed bounds on the condition numbers of the $L$ and $U$ factors, but we suspect that they can only grow polynomially, rather than exponentially, with dimension.

# 8 Diagonally Scaled Totally Unimodular (DSTU) matrices

The following definition is taken from [10]. A matrix $Z$ with integer entries is called *totally unimodular (TU)* if all of its minors are $-1$, 0 or $+1$. In particular, the entries of $Z$ must be $-1$, 0 or $+1$. We further define a matrix $G$ to be *diagonally scaled totally unimodular (DSTU)* if it can be written $G = D_L Z D_R$, where $Z$ is TU, and $D_L$ and $D_R$ are diagonal. In our applications $Z$ will be known exactly, but the diagonal entries of $D_L$ and $D_R$ will only be known to high relative accuracy. The determinant $\det(G) = \det(D_L) \cdot \det(Z) \cdot \det(D_R)$ is determined to high relative accuracy since $\det(Z)$ is known exactly, and the other two determinants are products of numbers known to high relative accuracy. Since any submatrix of a DSTU matrix is DSTU, all minors are determined to high relative accuracy, so all entries of the $L$, $D$ and $U$ factors of $G$ are determined to high relative accuracy.

A variety of characterizations of TU matrices are given in [10, sec. 2.3]. We limit ourselves to two examples: the acyclic matrices discussed in section 6, and the finite element matrices from linear mass-spring systems in section 12.1, which we discuss in that section. We also note that the *reduced note-arc incidence (RNAI) matrices* of [55] are a special case of TU matrices. In [17], we characterized acyclic sparsity patterns as follows:

**Theorem 8.1** Let $\mathcal{G}$ be the class of matrices with a given sparsity pattern. Let $Z \in \mathcal{G}$ be the unique matrix with all entries equal to 0 or 1. Then $\mathcal{G}$ is acyclic if and only if all matrices $G \in \mathcal{G}$ can be written $G = D_L Z D_R$ for some diagonal matrices $D_L$ and $D_R$.

Since $Z$ is acyclic too, each minor of $Z$ consists of at most one monomial, and so is $-1$, 0 or $+1$. Thus, $Z$ is TU and $G = D_L Z D_R$ is DSTU.

It remains to give an algorithm for performing GECP on a DSTU matrix, and to show that the $L$ and $U$ factors it computes are well-conditioned.

**Algorithm 8.1** Performing Gaussian elimination with pivoting $G = P_r L D U P_c$ in a forward stable manner on an $m$-by-$n$ DSTU matrix $G = D_L Z D_R$, where $m \geq n$.

```
    for i = 1 to min(m − 1, n)
        pivot so that G_ii is nonzero
        D_ii = G_ii
        for j = i + 1 to m
            L_ji = G_ji / D_ii
        endfor
        for j = i + 1 to m
            for k = i + 1 to n
(*)             G_jk = G_jk − L_ji · G_ik
```

```
(***)           If the last subtraction has two nonzero operands, set G_jk = 0
            endfor
        endfor
        for k = i+1 to n
            U_ik = G_ik/D_ii
        endfor
```

The above algorithm is essentially identical to conventional Gaussian elimination, except for line (***).

**Theorem 8.2** *Algorithm 8.1 computes all the entries of the L, D and U factors of a DSTU matrix G to high relative accuracy, for any pivot sequence not dividing by zero. If we use complete pivoting, and $m = n$, then the entries of $L^{-1}$ and $U^{-1}$ are bounded by 1 in absolute value, so that $\kappa(L) = O(n^2)$ and $\kappa(U) = O(n^2)$. If $m > n$, so L is m-by-n, then $\kappa(L) = O(mn)$. These bounds are attainable. In other words, L and U have condition numbers that grow at most quadratically with dimension.*

*Proof:* First we show that Algorithm 8.1 is forward stable for any pivot sequence not dividing by zero. Since floating point multiplication and division are forward stable (i.e. they compute the result to high relative accuracy if the operands are known to high relative accuracy), the only potential source of inaccuracy is the subtraction in line (*). We claim that the only situation in which $G_{jk}$ and $L_{ji} \cdot G_{ik}$ are both nonzero is when they are equal (in exact arithmetic), so the result is exactly zero; this situation is accounted for in line (***).

To see that line (***) in Algorithm 8.1 computes $G_{jk}$ exactly if the if-test is satisfied, we note that $G_{ij}$ before executing (*), $G_{jk}$ after executing (*), and $L_{ji} \cdot G_{ik}$ are all quotients of minors of $G$ (or products of quotients of minors of $G$), i.e. quotients of monomials in the variables $D_{L,ii}$ and $D_{R,jj}$, with coefficients $\pm 1$, or zero. (We call a monomial with coefficient $\pm 1$ a $\pm 1$-monomial for short.) Now think of (*) as a polynomial identity $m_1 = m_2 + m_3$ among $\pm 1$-monomials (or zero) $m_i$ in the variables $D_{L,ii}$, $D_{L,ii}^{-1}$, $D_{R,ii}$ and $D_{R,ii}^{-1}$. Then since all coefficients can only be 0 or $\pm 1$, the only way both $m_2$ and $m_3$ can be nonzero is if their sum $m_1$ cancels exactly to zero. Thus $m_1$ must be zero if $m_2$ and $m_3$ are nonzero. This completes the proof that Algorithm 8.1 is forward stable.

Next we show that with complete pivoting, the entries of $U^{-1}$ and $L^{-1}$ are bounded by one in absolute value if $m = n$. It suffices to consider $U$. We use the fact that any entry $U_{ij}^{-1}$ is both a quotient of minors of $G$, and a quotient of minors of $U$. Since $G$ is DSTU, its minors are $\pm 1$-monomials in the diagonal entries of $D_L$ and $D_R$, so we can write $U_{ij}^{-1} = \text{mono}_n(D_L, D_R)/\text{mono}_d(D_L, D_R)$. Since $U$ is a unit triangular matrix, its determinant is 1, so $U_{ij}^{-1}$ is really just a minor of $U$, which in turn is a sum $\sum_k \text{mono}_k(U)$, of $\pm 1$-monomials in the entries of $U$. Since all $|U_{kl}| \le 1$ by complete pivoting, $|\text{mono}_k(U)| \le 1$ too. We will show that at most one term $\text{mono}_k(U)$ in the sum for $U_{ij}^{-1} = \sum_k \text{mono}_k(U)$ is nonzero, implying that $|U_{ij}^{-1}| \le 1$ as desired.

To this end, note that each $\text{mono}_k(U) = \text{mono}_{n,k,U}(D_L, D_R)/\text{mono}_{d,k,U}(D_L, D_R)$ is the quotient of $\pm 1$-monomials of diagonal entries of $D_L$ and $D_R$, since each entry of $U$ is the quotient of minors of $G$, each of which is a $\pm 1$-monomial in $D_L$ and $D_R$. This implies

$$\frac{\text{mono}_n(D_L, D_R)}{\text{mono}_d(D_L, D_R)} = U_{ij}^{-1} = \sum_k \frac{\text{mono}_{n,k,U}(D_L, D_R)}{\text{mono}_{d,k,U}(D_L, D_R)} \quad .$$

Now think of this as a polynomial identity in the variables $D_{L,ii}$, $D^{-1}_{L,ii}$, $D_{R,ii}$ and $D^{-1}_{R,ii}$. This algebraic identity can only hold if there is exact cancellation among the monomials in the sum on the right, so that at most one term remains after cancellation. This completes the proof that $|U^{-1}_{ij}| \le 1$. Thus $\|U\| \le n$ and $\|U^{-1}\| \le n$ (in the 1, 2, or $\infty$ norm), and $\kappa(U) \le n^2$.

The same argument applies to $L$ when $L$ is square, and may be modified easily when $L$ is rectangular.

To see that the condition number $O(n^2)$ can be attained, consider the acyclic matrix $G$ defined by the following Matlab program:

$$
\begin{aligned}
&G = \mathrm{eye}(4 \cdot n + 1);\\
&G(1, 2 : n + 1) = \mathrm{ones}(1, n);\\
&G(n + 2 : 2 \cdot n + 1, 1) = \mathrm{ones}(n, 1);\\
&G(n + 2 : 3 \cdot n + 2, 4 \cdot n + 1) = \mathrm{ones}(1, n);\\
&G(2 \cdot n + 2 : 3 \cdot n + 1, n + 2) = \mathrm{ones}(n, 1);
\end{aligned}
$$

Then $L$, $U$, $L^{-1}$ and $U^{-1}$ all contain $n$-by-$n$ blocks of $+1$s (or $-1$s). □

To illustrate the phenomenon of exact cancellation, consider the (singular) acyclic matrix

$$G = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} .$$

If we choose $G_{11}$ as the pivot, then after one step the trailing 2-by-2 submatrix has each entry equal to $-1$. Choosing any of these entries as the next pivot causes exact cancellation in the third and final pivot, which is zero. This phenomenon can also occur with nonsingular acyclic matrices, such as

$$G = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} . \tag{25}$$

It is probably possible to implement Algorithm 8.1 in $o(n^3)$ time, but since the subsequent Algorithm 3.1 for computing the SVD itself takes $\Theta(n^3)$ operations, we have not pursued this. It remains to design a high accuracy algorithm for the singular vectors of an acyclic matrix taking $o(n^3)$ time.

We present numerical examples in section 12.2.

### 8.1 Accurate SVDs of other matrices of the form $G = D_L Z D_R$

In the last section we considered the case where $Z$ was TU, i.e. each minor of the integer matrix $Z$ was bounded by 1 in absolute value. It is natural to ask what happens when the bound on each minor is $\bar{k} > 1$. In this case one can still compute an accurate LDU decomposition of $G$, by performing GECP on $Z$ in rational arithmetic, and using $D_L$ and $D_R$ only for pivot selection (diagonally scaling each Schur complement of $Z$ just in order to choose the largest entry, but performing the actual elimination on $Z$ itself). The bound $\bar{k}$ on minors of $Z$ implies that any rational number $s/r$ in lowest terms appearing during Gaussian elimination on $Z$ has $s$ and $r$

bounded in magnitude by $\bar{k}$; this is why the TU case of $\bar{k} = 1$ was so easy. The argument that showed $|U_{ij}^{-1}| \le 1$ may be extended to show that $|U_{ij}^{-1}| \le \bar{k}^2$, whence $\kappa(U) \le (\bar{k}n)^2$. We do not know if this is attainable for $\bar{k} > 1$.

# 9 Cauchy Matrices

In this section and the next we consider matrices whose entries are rational functions of a number of parameters. These matrices will have the property that expressions for their minors exist, that can be evaluated to high relative accuracy when the parameters are given floating point numbers. This then determines their $LDU$ factors to high relative accuracy, and so their SVDs to high relative accuracy.

Cauchy matrices are defined to have entries $C_{i,j} = 1/(x_i + y_j)$, where the $x_i$ and $y_j$ are the initial data. Every submatrix of a Cauchy matrix is Cauchy. The well-known formula for $\det(C)$ is

$$\det(C) = \frac{\prod_{1 \le i < j \le n}(x_j - x_i)(y_j - y_i)}{\prod_{1 \le i,j \le n}(x_i + y_j)} \tag{26}$$

Every factor $x_j - x_i$, $y_j - y_i$, or $x_i + y_j$ is computed to high relative accuracy, as are their products and quotients, for the reasons discussed in section 5.

For example, consider the Hilbert matrix, where $x_i = i$ and $y_j = j - 1$. When $n = 13$, the determinant as computed by Gaussian elimination with no pivoting, partial pivoting or complete pivoting has lost all relative accuracy compared to the true value of the above formula, of about $1.44 \cdot 10^{-92}$.

Note that small relative changes in the $x_i$ and $y_j$ do *not* necessarily guarantee small relative changes in $\det(C)$, as the 1-by-1 example with $x_1 = -1$ and $y_1 = 1 + \epsilon$ shows. But since formulas like (26) can be evaluated to high relative accuracy, they simply give the right answer, independent of conditioning; all significant errors occur when $x_i$ and $y_j$ are rounded to floating point numbers. But if there is little cancellation in the factors $x_j - x_i$, $y_j - y_i$ and $x_i + y_j$ (i.e. they are close enough in magnitude to $|x_j| + |x_i|$, $|y_j| + |y_i|$ and $|x_i| + |y_j|$, respectively), then one can assert that small relative changes in the $x_i$ and $y_j$ cause small relative changes in the SVD. This is true for the Hilbert matrix, for example. (The Hilbert matrix is also *totally positive*, as discussed in the next section.)

The amount of work required to compute the $LDU$ factorization with formula (26) is much larger than the work required for straightforward Gaussian elimination. Its most straightforward use would cost $O(n^5)$, although a dynamic programming approach could probably reduce this. So finding a really practical way to do high accuracy GECP remains open. Still, formula (26) shows that high accuracy GECP is achievable in principle.

# 10 Totally Positive Matrices

A matrix is *totally positive (TP)* if all of its minors are nonnegative [35]. This suggests that there should be formulas for minors that somehow automatically guarantee positivity, and so high relative accuracy. However, total positivity alone is not enough to guarantee that GECP can be performed accurately. For example, the Hilbert matrix is TP, but unless we exploit further information about the matrix, such as it being Cauchy, we do not expect straightforward GECP to be accurate enough.

Similarly, a symmetric tridiagonal matrix $T$ with positive offdiagonal entries is totally positive if and only if it is positive definite [35, Thm. 3.2]. Simply knowing the entries of $T$ to high relative accuracy does not determine the SVD to high relative accuracy, but knowing the entries of $T$'s bidiagonal Cholesky factor to high relative accuracy does. So achieving high relative accuracy requires not just total positivity but an appropriate parameterization that permits minors to be evaluated to high relative accuracy. We give many examples of this below.

The rest of this section is organized as follows. First, we give several examples of TP matrices and their parameterizations where high accuracy formulas for their minors exist [2, 9, 8, 35]. Indeed, it was recently shown [9] that there is a universal parameterization of all totally positive matrices with this property, although this parameterization is not always convenient to use. Second, we show that well known *composition laws* for producing new TP matrices from previous ones also produce new high accuracy formulas from previous ones. This can be used to generate many matrices for which high accuracy formulas exist. Unfortunately, the formulas we present are often combinatorially expensive, so they are not always practical for large problems. Still, they show that high relative accuracy is achievable, and motivate us to seek more economical formulas for problems of particular interest.

### 10.1 Examples of Totally Positive Matrices with High Accuracy Formulas for their Minors

1. *Cauchy matrices* are TP, provided the $x_i$ and $y_j$ in $C_{ij} = 1/(x_i + y_j)$ satisfy $0 < x_1 < x_2 < \cdots < x_n$ and $0 < y_1 < y_2 < \cdots < y_n$. Formula (26) can be used to compute minors to high relative accuracy.

2. An $n$-by-$n$ *generalized Vandermonde matrix* has entries $V_{i,j} = z_j^{e_i}$, where $0 \le e_1 < e_2 < \cdots < e_n$ are given integers. The usual Vandermonde matrix is given by $e_i = i - 1$. Every submatrix of a generalized Vandermonde matrix is also generalized Vandermonde. The formula for $\det(V)$ is [38, eqn. 3.1]

$$\det(V) = [\prod_{1 \le i < j \le n} (z_j - z_i)] \cdot s_\mu(z_1, ..., z_n) \tag{27}$$

where $s_\mu$ is the so-called *Schur function*, and $\mu$ is the *partition* $\mu = (e_n - (n-1), e_{n-1} - (n-2), ..., e_2 - 1, e_1)$. A great deal is known about Schur functions [38], but for the purposes of showing that GECP can in principle be implemented very accurately, all we need to know is that if we write $s_\mu$ as a sum of monomials [38, p. 73]

$$s_\mu = \sum_i [\alpha_i z_1^{\beta_{i1}} z_2^{\beta_{i2}} \cdots z_n^{\beta_{in}}] \tag{28}$$

then all its nonzero coefficients $\alpha_i$ are positive integers (see [38, p. 73] for a combinatorial formula for the $\alpha_i$ and $\beta_{i,j}$). Therefore, if the $z_j$ satisfy $0 < z_1 < z_2 < \cdots < z_n$, equations (27) and (28) tell us that $V$ is TP, and provide us a formula for $\det(V)$ that can be evaluated to high relative accuracy. Unfortunately, the number of monomials in the Schur function grows very quickly as a function of $\mu$, so this is not a practical formulas for large matrices. There are alternate formulas for Schur functions, as *Jacobi-Trudi* determinants [38], but these appear to be no easier to evaluate to high relative accuracy than the original problem.

3. *Upper triangular Toeplitz matrices* $T$ with certain special forms are TP. The most basic ones (from which we build others below) are bidiagonal matrices with 1 on the diagonal and first superdiagonal (each minor is 0 or 1), matrices with 1 on and everywhere above the diagonal (each minor is 0 or 1), and the Taylor matrix $T_{ij} = 1/(j-i)!$ (each minor is rational, and can in principle be evaluated exactly using rational Gaussian elimination).

4. Brenti [9, Thm. 3.1.] [8] has recently shown that there is a one-to-one correspondence between totally positive matrices $T$ and planar, finite, nonnegatively edge-labeled directed graphs, with certain distinguished *row nodes* and *column nodes*. Given such a graph, $T_{ij}$ is given as a sum, over all paths $\pi$ from row node $i$ to column node $j$, of the product of the edge weights along $\pi$. If the edge weights are known to high relative accuracy, this sum is determined to high relative accuracy. Furthermore, any $r$-by-$r$ minor of $T$ can also be defined as a sum over certain $r$-tuples of nonintersecting paths of products of edge weights, which again is determined to high relative accuracy.

   The proof of Brenti's theorem involves the construction of an appropriate graph given a TP matrix $T$. This construction is nothing other than Gaussian elimination [15], with the restriction of eliminating using only adjacent rows or columns, thus expressing $T$ as a product of (TP) Gauss transforms, which differ from a diagonal matrix by only one entry of the first super- or subdiagonal, and (TP) shifts. In other words, the problem of building the desired graph to high relative accuracy is equivalent to the problem we wanted to solve in the first place, Gaussian elimination to high relative accuracy[5].

   Brenti [9, Thm. 3.3.] also showed that there is a "universal graph" for all $n$-by-$n$ TP matrices, where the edge weights are arbitrary nonnegative reals. In other words, these nonnegative reals parameterize the set of all $n$-by-$n$ TP matrices. And given these parameters to high relative accuracy, the graph provides a way to compute any minor to high relative accuracy. Again, computing these parameters from a TP matrix $T$ is done with a variation of Gaussian elimination.

### 10.2 Composition Laws for Totally Positive Matrices

1. If $A$ is TP with high accuracy formulas for all minors, then the same is true of $A^T$.

2. If $A$ and $B$ are $m$-by-$r$ and $r$-by-$n$ TP matrices, respectively, their $m$-by-$n$ product $AB$ is also TP. This follows from the Cauchy-Binet Theorem, which expresses each $k$-by-$k$ minor of $AB$ as a sum of $\binom{r}{k}$ products of minors of $A$ and of $B$. If we have high accuracy formulas for all minors of $A$ and $B$, Cauchy-Binet also gives us a high accuracy formula for all minors of $AB$, as a sum of products of positive minors, each of which is computable to high relative accuracy.

   **Example 10.1** Suppose $X$ and $Y$ are TP Vandermonde with entries $X_{ji} = x_j^{i-1}$ and $Y_{ji} = y_j^{i-1}$ respectively, and $D = \mathrm{diag}(d_1, ..., d_n)$ with $d_i > 0$, then $P = X^T DY$ is TP, with entries $P_{ij} = \sum_{k=1}^n d_k x_i^{k-1} y_j^{k-1}$. In particular, suppose $0 < x_0 < x_1$, $0 < y_0 < y_1$, and $d_k = 1/(k-1)!$;

[5] This is yet another example of the "no free lunch" principle in getting error bounds.

then

$$
\begin{aligned}
\det(P) &\equiv \det\begin{bmatrix} e^{x_0y_0} & e^{x_0y_1} \\ e^{x_1y_0} & e^{x_1y_1} \end{bmatrix} \\
&= \det\left(\begin{bmatrix} 1 & x_0 & x_0^2 & \cdots \\ 1 & x_1 & x_1^2 & \cdots \end{bmatrix} \cdot \mathrm{diag}\left(\frac{1}{0!},\frac{1}{1!},\frac{1}{2!},\cdots\right) \cdot \begin{bmatrix} 1 & y_0 & y_0^2 & \cdots \\ 1 & y_1 & y_1^2 & \cdots \end{bmatrix}^T\right) \\
&= \sum_{0\le i<j} \det\left(\begin{bmatrix} x_0^i & x_0^j \\ x_1^i & x_1^j \end{bmatrix} \cdot \mathrm{diag}\left(\frac{1}{i!},\frac{1}{j!}\right) \cdot \begin{bmatrix} y_0^i & y_0^j \\ y_1^i & y_1^j \end{bmatrix}^T\right) \\
&= (x_1-x_0)\cdot(y_1-y_0)\cdot \sum_{0\le i<j}\left(\frac{(x_0x_1y_0y_1)^i}{i!\cdot j!}\cdot \sum_{k=0}^{j-i-1} x_0^k x_1^{j-i-1-k} \cdot \sum_{k=0}^{j-i-1} y_0^k y_1^{j-i-1-k}\right) ,
\end{aligned}
$$

all of whose factors are evaluatable to high relative accuracy, for the reasons discussed in section 5.

**Example 10.2** As another example, consider $D_1AD_2$, where $A$ is TP and Toeplitz, and $D_1$ and $D_2$ are positive definite diagonal with $D_{1,ii} = \alpha^{1-i}$, and $D_{2,ii} = \alpha^{i-1}$. Then $D_1AD_2$ is also a TP Toeplitz matrix, with the $i$-th diagonal multiplied by $\alpha^i$. Any minor of $D_1AD_2$ is equal to the corresponding minor of $A$ times an appropriate power of $\alpha$.

3. Let $A$ be TP and $S = \mathrm{diag}(+1,-1,+1,-1,...)$. Then $SA^{-1}S$ is also TP, and every minor of $SA^{-1}S$ is a quotient of minors of $A$. In other words, $A^{-1}$ has a checkerboard sign pattern. For example, if $A$ is upper bidiagonal Toeplitz with 1 on the diagonal and first superdiagonal, then $SA^{-1}S$ is upper triangular Toeplitz with all ones on and above the diagonal.

4. Theorems of Aissen, Schoenberg, Whitney; Whitney; and Erdrei [2, p. 215] show that *all* totally positive Toeplitz matrices can be assembled from the above operations applied to the basic TP Toeplitz matrices mentioned earlier. A row of an upper triangular totally positive Toeplitz matrix is called a *Pólya frequency sequence* [35, 2].

5. If $A$ is TP, so is any Schur complement of $A$. Since any minor of the Schur complement is a quotient of minors of $A$, high accuracy formulas for minors of $A$ yield high accuracy formulas for minors of the Schur complement.

6. Let $A(x_1, x_2, ...x_p)$ be a TP matrix, when $0 < x_1 < x_2 < \cdots < x_p$. Then if $\phi(x)$ is a nonnegative strictly increasing function, $B = A(\phi(x_1), \phi(x_2), ..., \phi(x_p))$ is also TP. To get a high accuracy formula for minors of $B$ from the corresponding formula for minors of $A$, we require (1) that $\phi(x)$ can be evaluated to high relative accuracy if $x$ is given to high relative accuracy, and (2) that $\phi(x) - \phi(y)$ can be evaluated to high relative accuracy, if $x$ and $y$ are floating point numbers. For example, if $\phi(x)$ is a polynomial in $x$ with nonnegative coefficients, it satisfies these conditions.

## 11 Solving other linear algebra problems with high relative accuracy

It is natural to ask when other linear algebra problems have solutions determined to high relative accuracy, given the combinatorial and algebraic properties of previous sections. We consider matrix inversion, computing the QR factorization, and (more generally) solving least squares problems.

First we consider computing $G^{-1}$. Since each entry of $G^{-1}$ is $\pm 1$ times a quotient of an $n-1$-by-$n-1$ minor of $G$ and $\det(G)$, we only need these $n^2+1$ largest minors to be determined accurately. Then it is immediate that all our earlier conditions that imply that *all* minors are determined accurately also imply that all entries of the inverse are determined accurately.

For example, previous authors have noted that a linear system with a Vandermonde coefficient matrix $V$ can be solved quite accurately precisely when it is TP, although only some authors used this language [32, 5, 7, 6]. This high accuracy phenomenon is now understandable, since linear system solving can also be expressed in terms of minors, and should apply to all linear systems with TP coefficient matrices. It is worth noting that the standard fast algorithm for Vandermonde systems can be described as providing a factorization of a TP Vandermonde into a product of simpler TP matrices, each of which has simple high accuracy formulas for all its minors (this factorization, which applies to non-TP Vandermondes, appears in [28]), thus providing another high accuracy way to evaluate Schur functions [38]. But fast algorithms remain hard to design. Perhaps the successes in divide and conquer algorithms for TP linear system solving can be translated into similar algorithms for the pivoted LDU decomposition and so SVD.

Since linear equation requires only that the $n^2+1$ largest minors be determined accurately, it is possible to compute an accurate inverse more often than an accurate SVD. If the Laplace expansions of the $n^2+1$ largest minors of $G$ are sums of monomials of like sign, so that each entry of $G^{-1}$ is determined to high relative accuracy, then Brualdi and Shader call $G$ *strongly sign nonsingular*, or $S^2NS$ [11]. To see that these matrices form a strictly larger class than either acyclic or TSC matrices, consider

$$G = \begin{bmatrix} D_1 & X \\ 0 & D_2 \end{bmatrix} \quad , \quad G^{-1} = \begin{bmatrix} D_1^{-1} & -D_1^{-1} X D_2^{-1} \\ 0 & D_2^{-1} \end{bmatrix}$$

where $D_1$ and $D_2$ are nonsingular diagonal matrices, and $X$ is arbitrary. $G^{-1}$ is determined to high relative accuracy, even though $G$ may be neither acyclic nor TSC, depending on $X$. In other words, determining the inverse to high relative accuracy is strictly "easier" than computing either the LU decomposition or SVD to high relative accuracy. See [11] for further discussion.

Now we consider the QR factorization $G = QR$ and least squares problems $\min_x \|Gx - b\|_2$. It is natural to expect similar high accuracy results as before, because of the following well-known facts:

- $R^T R$ is the Cholesky factorization of $G^T G$, so that each entry of $R$ is a (quotient of) square root(s) of minor(s) of $G^T G$.
- $QQ^T = G(G^T G)^{-1} G^T$, so if $G^T G$ is accurately invertible, we expect that $Q$ might be determined accurately.
- The pseudoinverse $G^+ = (G^T G)^{-1} G^T$, so again we expect this might be accurate if $G^T G$ is accurately invertible.

For example, it is natural to conjecture that a theorem like Theorem 6.1 is true for the QR decomposition. Indeed, we believe that the entries of $R$ in $G = QR$ are determined to high relative accuracy for all $G \in \mathcal{G}$ if and only if $\mathrm{Gr}(\mathcal{G})$ is acyclic, but we have not been able to prove this. It also seems likely that the Householder vectors determining $Q$ are determined to high accuracy.

On the other hand, being TSC is definitely not enough to guarantee an accurate QR decomposition. For example, consider $G = \begin{bmatrix} 1 & 1 \\ 1 & -1-\epsilon \end{bmatrix}$; there is unavoidable cancellation in computing $R_{12}$. However, the diagonal entries of $R$ are determined to high relative accuracy if $G$ is TSC. This follows from thinking of $R$ as the Cholesky factor of $G^TG$, and the diagonal entries of $R$ as square roots of quotients of principal minors of $G^TG$. By the Cauchy-Binet theorem, a principal minor of $G^TG$ can be written as a sum of squares of minors of $G$, each of which is determined to high relative accuracy. This also implies that $G^TG$ is accurately invertible if $G$ is TP, although there will be cancellation in the products defining $QQ^T$ and $G^+$ above, since $(G^TG)^{-1}$ will have a checkerboard sign pattern.

It remains to characterize those matrices whose QR factorization, and associated least squares problems, can be solved to high relative accuracy.

## 12 Finite Element Problems

As described earlier, the most natural finite element formulation leads to a generalized eigenproblem of the form $Kx = \lambda Mx$, where $M$ is the *mass matrix*, and $K$ is the *stiffness matrix*. Typically we write $K = Z_K^T D_K Z_K$ where $Z_K$ is the *incidence matrix* or *assembly matrix*, and $D_K$ is the (block) diagonal matrix of individual element stiffnesses. We may similarly write $M = Z_M^T D_M Z_M$. Assume first for simplicity that $Z_M$ is square and invertible, and that $D_M$ and $D_K$ are positive definite diagonal. Then we can pre- and postmultiply the eigenproblem $K - \lambda M$ by $D_M^{-1/2} Z_M^{-T}$ and $Z_M^{-1} D_M^{-1/2}$, respectively, to get the eigenproblem $G^TG - \lambda I$, where

$$G = D_K^{1/2}(Z_K Z_M^{-1}) D_M^{-1/2} \equiv D_1 B D_2 \ .$$

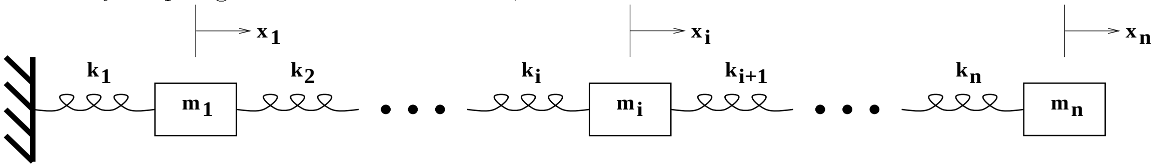
Thus, the problem reduces to finding the SVD of $G = D_1 B D_2$.

We can think of $G$ as the "unassembled" finite element problem. The diagonal matrices $D_1 = D_K^{1/2}$ and $D_2 = D_M^{-1/2}$ depend only on the *material properties* of the finite element problem (such as masses and spring constants in the example below), whereas $B = Z_K Z_M^{-1}$ depends only on the *geometry and meshing*. In the examples we have studied, the (worst case) relative accuracy of the SVD of $G$ depends *only* on $B$, i.e. the geometry and meshing, *not* on the material properties in $D_1$ and $D_2$. This is because $D_1$ and $D_2$ affect the pivot choice during GECP, but the accuracy depends only on the conditioning of submatrices of $B$. In contrast, we will show that the conventional assembled problem $K - \lambda M$ may unavoidably destroy high relative accuracy. A similar phenomenon in the case of linear systems was analyzed in [55, 54].

In the more general case where $Z_M$ is not square and invertible, we would need to compute the generalized SVD (GSVD) of the pair $(D_K^{1/2} Z_K, D_M^{1/2} Z_M)$. Many of the perturbation theorems and algorithms of this paper may be extended to this case. For example, we can show that a conventional piecewise linear discretization of the Sturm-Liouville problem $-(a(x)y')' = \lambda\rho(x)y$ on $0 < x < 1$ with rather general boundary conditions leads to a GSVD formulation that determines its small modes of vibration to high relative accuracy [23], and for which we have an algorithm that depends on material in this paper and in [22]. We will report on these and other finite element problems in more detail in a future paper.

### 12.1 Linear Mass Spring Systems

A *linear mass spring system* consists of masses $m_i$ that may move only horizontally, and springs with spring constants $k_l$ connecting arbitrary pairs of masses $m_i$ and $m_j$. Each mass may also be connected by a spring to an immovable wall, as shown below:




$x_i$ = position of i-th mass (0 = equilibrium)
$m_i$ = mass of i-th mass
$k_i$ = spring constant of i-th spring

Newton's Law applied to the system in the figure yields $M\ddot{x}(t) = -Kx(t)$, where $x(t) = [x_1(t), ..., x_n(t)]^T$, $M = \text{diag}(m_1, ..., m_n)$, and

$$K = \begin{bmatrix} k_1+k_2 & -k_2 & & & \\ -k_2 & k_2+k_3 & -k_3 & & \\ & \ddots & \ddots & \ddots & \\ & & -k_{n-1} & k_{n-1}+k_n & -k_n \\ & & & -k_n & k_n \end{bmatrix} .$$

Seeking solutions of the form $x(t) = e^{\mu t}x$ leads to the usual generalized eigenproblem $Kx = \mu^2 Mx \equiv \lambda Mx$. We will see below that this formulation does *not* preserve high relative accuracy in the eigenvalues.

To reformulate the problem so that high relative accuracy is preserved, we write $K = Z_K^T D_K Z_K$ where $D_K = \text{diag}(k_1, ..., k_n)$ and $Z_K$ is the bidiagonal *incidence matrix*

$$Z_K = \begin{bmatrix} 1 & & & \\ -1 & \ddots & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix} \tag{29}$$

where the nonzero $z_{K,ij}$ indicates that spring $i$ is connected to mass $j$. Furthermore, $M = Z_M^T D_M Z_M$ where $D_M = \text{diag}(m_1, ..., m_n)$ and $Z_M = I$, so our problem reduces to finding the SVD of $G = D_K^{1/2} Z_K D_M^{-1/2}$, which is a bidiagonal matrix with entries $G_{i,i} = \sqrt{k_i/m_i}$ and $G_{i+1,i} = -\sqrt{k_{i+1}/m_i}$. The singular values of $G$ are the square roots of the eigenvalues. Each entry of $G$ is determined to about as many decimal places as the spring constants $k_i$ and masses $m_i$. $G$ is bidiagonal, and so acyclic, and so its singular values are determined to about as many decimal places as the data, which is as accurate an answer as the data deserves.

More generally, when there are springs between arbitrary pairs of masses, or arbitrary masses and the wall, $Z_K$ will have the following structure. Row $i$ will have $-1$ and $+1$ in columns $j$ and

$k$, respectively, if spring $i$ connects masses $j$ and $k$. Row $i$ will have $\pm 1$ in column $j$ if spring $i$ connects mass $j$ to the wall. It was shown by Hoffman and Kruskal that $Z_K$ is *totally unimodular (TU)* [10, Thm. 2.3.3.], although if no masses are connected to a wall then $Z_K$ is called an *oriented incidence matrix* and the result goes back to Poincaré. Thus, $G = D_K^{1/2} Z_K D_M^{-1/2}$ is a *diagonally scaled unimodular matrix (DSTU)*, and its SVD can be computed to high relative accuracy as described in section 8.

Now we show that the conventional assembled formulation $Kx = \lambda Mx$ does *not* necessarily preserve high relative accuracy, when the $k_i$ and $m_i$ have widely varying magnitudes. First, accuracy can be lost by a conventional algorithm like divide-and-conquer that only guarantees high absolute accuracy in the computed eigenvalues. Second, and independently of the algorithm used to solve the eigenproblem, accuracy can be lost simply by forming *and rounding* $K$ from the data $k_i$. For example, Suppose $n = 3$, $M = I$, $k_1 = k_3 = 1$, and $k_2 = \varepsilon/2$ (so $\mathrm{fl}(k_1 + k_2) = k_1$). Then the correctly rounded $K$ is

$$\mathrm{fl}(K) = \begin{bmatrix} 1 & -\varepsilon/2 & 0 \\ -\varepsilon/2 & 1 & -1 \\ 0 & -1 & 1 \end{bmatrix}$$

This matrix is easily seen to have a tiny negative eigenvalue near $-\varepsilon^2/8$, whereas its true tiniest eigenvalue must be positive, in fact near $\varepsilon/4$ (Matlab returns 0 in place of $-\varepsilon^2/8$).

## 12.2 Numerical Examples

In this example, we apply Algorithm 8.1, GECP for DSTU matrices, followed by Algorithm 3.1, to several linear mass-spring systems, and compare the results to several other algorithms. These examples will be rank deficient.

We generate test matrices of the type $G = D_1 B D_2$, where

$$B = \begin{bmatrix} 1 & & & -1 & & \\ & 1 & & & -1 & \\ & & 1 & & -1 & \\ 1 & -1 & & & & \\ & 1 & -1 & & & \\ & & 1 & & & -1 \end{bmatrix}$$

and $D_1$ and $D_2$ are diagonal with $\kappa(D_1) = 10^i$, $\kappa(D_2) = 10^j$, $1 \le i, j \le 8$. For each fixed $(i, j)$ we generate 90 test matrices, using 9 different types of distributions of the singular values of $D_1$ and $D_2$, for a total of $8 \cdot 8 \cdot 90 = 5760$ test matrices.

The algorithms tested are as follows:

- Algorithm 8.1 (GECP on DSTU matrices) followed by Algorithm 3.1. We call the combined algorithm `SLUSVD` in single precision and `DLUSVD` in double precision).

- One-sided Jacobi in single precision (`SGSVDJ`) and double precision (`DGSVDJ`).

- QR based SVD in single precision from LAPACK `SGESVD`.

Since in this case only the first, LU based method can determine the rank exactly, we measure the relative error only in the $\mathrm{rank}(G) = 5$ nonzero singular values. We use $\varepsilon(i, j)$ to denote the

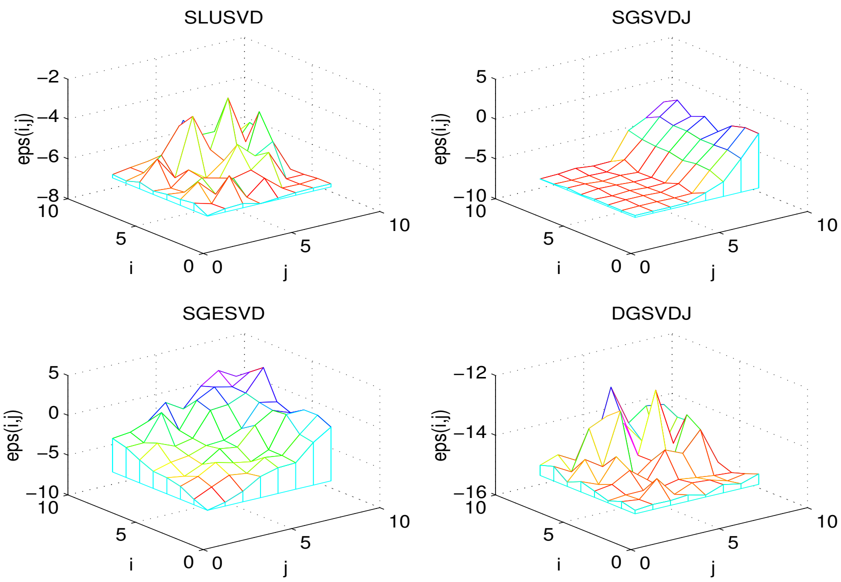



Figure 2: The values of of $\log_{10} \varepsilon(i,j)$ for $1 \le i, j \le 8$.

maximal relative error in nonzero singular values of all test matrices with fixed $\kappa(D_1) = 10^i$ and $\kappa(D_2) = 10^j$. As a reference, we use the singular values computed by `DLUSVD`.

The measured values of $\varepsilon(i,j)$ for various algorithms are given in Figure 2. This figure shows that the single precision algorithm `SLUSVD` delivers at least 5 correct digits in all cases, whereas one-sided Jacobi or QR in single precision can lose all relative accuracy. One-sided Jacobi in double precision always delivers at least 13 decimal digits, although theory does not guarantee this.

## 13 Open Problems

We listed a number of open problems throughout the paper. We reiterate the most important ones here.

1. Several matrix classes we introduced required expensive variations on GECP to compute accurate LDU factorizations: TSC matrices cost $O(n^4)$, Cauchy matrices cost $O(n^5)$, and totally positive matrices could be exponential in $n$. It is desirable to have faster algorithms in all these cases. A natural question is the "subtraction-free" complexity of computing a Schur function, as discussed in section 10.1.

2. We only discussed finite element problems that could be reduced to the SVD of a single matrix. But the most general case involves the generalized SVD of two matrices. We have studied this for two-dimensional trusses, Sturm-Liouville problems, and have made some progress with more general cases. But a complete analysis remains to be done. Such an analysis would start with any continuous problem that determined its smallest eigenvalues to high relative accuracy, and then describe the finite element discretizations preserving this accuracy, along with algorithms to compute them this accurately.

3. We have described mostly dense matrix algorithms in this paper, costing $O(n^3)$ and sometimes more. Large eigenproblems typically require iterative methods (such as Lanczos) to compute a few eigenvalues at a reasonable cost. It would be desirable to identify matrix classes and inexpensive iterative algorithms that preserve high relative accuracy.

## Acknowledgements

We are indebted to Alan Edelman and Sergei Fomin at MIT for pointing out numerous relevant references, and Sergei Fomin in particular for several useful discussions.

## References

[1] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, and D. Sorensen. *LAPACK Users' Guide (second edition)*. SIAM, Philadelphia, 1995. 324 pages.

[2] T. Ando. Totally positive matrices. *Lin. Alg. Appl.*, 90:165–219, 1987.

[3] J. Barlow and J. Demmel. Computing accurate eigensystems of scaled diagonally dominant matrices. *SIAM J. Num. Anal.*, 27(3):762–791, June 1990.

[4] C. Bischof and G. Quintana-Orti. Computing rank-revealing QR factorizations of dense matrices. Argonne Preprint ANL-MCS-P559-0196, Argonne National Laboratory, 1996.

[5] Å Björck and V. Pereyra. Solution of Vandermonde systems of equations. *Math. Comp.*, 24(112):893–903, 1970.

[6] T. Boros, T. Kailath, and V. Olshevsky. Predictive partial pivoting and backward stability of fast Cauchy solvers. http://WWW-ISL.Stanford.EDU/people/olshevsk/papers.html, 1994.

[7] T. Boros, T. Kailath, and V. Olshevsky. The fast Björck-Pereyra-type algorithm for parallel solution of Cauchy linear equations. http://WWW-ISL.Stanford.EDU/people/olshevsk/papers.html, 1995.

[8] F. Brenti. The applications of total positivity to combinatorics, and conversely. Mathematics dept. report, Universitá degli Studi di Perugia, Perugia, Italy, 1996.

[9] F. Brenti. Combinatorics and total positivity. Mathematics Dept. Report, Universitá degli Studi di Perugia, Perugia, Italy, 1996.

[10] R. Brualdi and H. Ryser. *Combinatorial Matrix Theory*. Cambridge University Press, 1991.

[11] R. Brualdi and B. Shader. *Matrices of sign-solvable linear systems*. Cambridge University Press, 1995.

[12] P. A. Businger and G. Golub. Algorithm 358: Singular value decomposition of a complex matrix. *Comm. Assoc. Comput. Mach.*, 12:564–565, 1969.

[13] T. Chan. Rank revealing QR factorizations. *Lin. Alg. Appl.*, 88/89:67–82, 1987.

[14] S. Chandrasekaran and I. Ipsen. On rank-revealing QR factorizations. *SIAM Journal on Matrix Analysis and Applications*, 15, 1994.

[15] C. Cryer. Some properties of totally positive matrices. *Lin. Alg. Appl.*, 15:1–25, 1976.

[16] P. Deift, J. Demmel, L.-C. Li, and C. Tomei. The bidiagonal singular values decomposition and Hamiltonian mechanics. *SIAM J. Num. Anal.*, 28(5):1463–1516, October 1991. (LAPACK Working Note #11).

[17] J. Demmel and W. Gragg. On computing accurate singular values and eigenvalues of acyclic matrices. *Lin. Alg. Appl.*, 185:203–218, 1993.

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

[19] J. Demmel and K. Veselić. Jacobi's method is more accurate than QR. *SIAM J. Mat. Anal. Appl.*, 13(4):1204–1246, 1992. (also LAPACK Working Note #15).

[20] J. Dongarra, J. Du Croz, I. Duff, and S. Hammarling. A set of Level 3 Basic Linear Algebra Subprograms. *ACM Trans. Math. Soft.*, 16(1):1–17, March 1990.

[21] Z. Drmač. *Computing the Singular and the Generalized Singular Values*. PhD thesis, Fernuniversität - Hagen, Hagen, Germany, 1994.

[22] Z. Drmač. Accurate computation of the product induced singular value decomposition with applications. Tech Report CU-CS-816-96, Computer Science Dept., University of Colorado at Boulder, July 1995. submitted to SINUM.

[23] Z. Drmač and K. Veselić. Sturm-Liouville operator discretized. preprint, 1996.

[24] S. Eisenstat and I. Ipsen. Relative perturbation techniques for singular value problems. *SIAM J. Numer. Anal.*, 32(6), 1995.

[25] K. Fernando and B. Parlett. Accurate singular values and differential qd algorithms. *Numerische Mathematik*, 67:191–229, 1994.

[26] K. Gates. private communication, 1996.

[27] S. K. Godunov, A. G. Antonov, O. P. Kirilyuk, and V. I. Kostin. *Guaranteed Accuracy in Numerical Linear Algebra*. Kluwer Academic Pubishers, 1993.

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

[29] M. Gu and S. Eisenstat. Relative perturbation theory for eigenvalues. Computer Science Dept. Report YALEU/DCS/RR-934, Yale University, February 1993.

[30] M. Gu and S. Eisenstat. An efficient algorithm for computing a strong rank–revealing QR factorization. *SIAM J. Sci. Comput.*, 17(4):848 – 869, 1996.

[31] M. Gu and S. C. Eisenstat. A divide-and-conquer algorithm for the bidiagonal SVD. *SIAM J. Mat. Anal. Appl.*, 16(1):79–92, January 1995.

[32] N. J. Higham. *Accuracy and stability of numerical algorithms.* SIAM, 1996.

[33] P. Hong and C. T. Pan. The rank revealing QR and SVD. *Math. Comp.*, 58:575–232, 1992.

[34] P. Hough and S. Vavasis. Complete orthogonal decomposition for weighted least squares. Computer Science Department Report, Cornell University, Ithaca, NY, March 1995.

[35] S. Karlin. *Total Positivity.* Stanford University Press, 1968.

[36] R.-C. Li. Relative Perturbation Theory: (i) Eigenvalue Variations. Computer Science Dept. Technical Report CS-94-252, University of Tennessee, Knoxville, September 1994. (LAPACK Working Note #84).

[37] R.-C. Li. Relative Perturbation Theory: (ii) Eigenspace Variations. Computer Science Dept. Technical Report CS-94-253, University of Tennessee, Knoxville, September 1994. (LAPACK Working Note #85).

[38] I. G. MacDonald. *Symmetric functions and Hall polynomials.* Oxford University Press, 2nd edition, 1995.

[39] R. Mathias. Accurate eigensystem computations by Jacobi methods. *SIAM J. Mat. Anal. Appl.*, 16(3):977–1003, 1996.

[40] D. O'Leary. On bounds for scaled projections and pseudoinverses. *Lin. Alg. Appl.*, 132:115–117, 1990.

[41] C. T. Pan. On the existence and computation of rank-revealing lu factorizations. submitted to Math. Comp., 1996.

[42] C. T. Pan and P. Tang. Bounds on singular values revealed by QR factorization. Technical Report MCS-P332-1092, Mathematics and Computer Science Division, Argonne National Laboratory, 1992.

[43] B. Parlett. *The Symmetric Eigenvalue Problem.* Prentice Hall, Englewood Cliffs, NJ, 1980.

[44] D. J. Pierce. Improved bound for Rank Revealing LU factorizations. preprint, 1993.

[45] E. Pietzsch. *Genaue Eigenwertberechnung nichtsingulärer schiefsymmetrischer Matrizen.* PhD thesis, Fernuniversität - Hagen, Hagen, Germany, 1993.

[46] R. A. Rosanoff, J. F. Gloudeman, and S. Levy. Numerical conditions of stiffness matrix formulations for frame structures. In *Proc.of the Second Conference on Matrix Methods in Structural Mechanics*, WPAFB Dayton, Ohio, 1968.

[47] B. Shader. private communication, 1995.

[48] I. Slapničar. *Accurate symmetric eigenreduction by a Jacobi method.* PhD thesis, Fernuniversität - Hagen, Hagen, Germany, 1992.

[49] I. Slapničar and K. Veselić. Perturbations of the eigenprojections of a factorized Hermitian matrix. *Lin. Alg. Appl.*, 218:274–280, 1995.

[50] G. W. Stewart. On scaled projections and pseudoinverses. *Lin. Alg. Appl.*, 112:189–193, 1989.

[51] G. W. Stewart. Updating a rank-revealing ULV decomposition. *SIAM J. Mat. Anal. Appl.*, 14(2):494–499, April 1993.

[52] J. G. Sun. Componentwise perturbation bounds for some matrix decompositions. *BIT*, 32:702–714, 1992.

[53] A. Van Der Sluis. Condition numbers and equilibration of matrices. *Num. Math.*, 14:14–23, 1969.

[54] S. Vavasis. Stable finite elements for problems with wild coefficients. *SIAM J. Num. Anal.*, ?:?, ?

[55] S. Vavasis. Stable numerical algorithms for equilibrium systems. *SIAM J. Mat. Anal. Appl.*, 15:1108–1131, 1994.

[56] K. Veselić. A Jacobi eigenreduction algorithm for definite matrix pairs. *Num. Math.*, 64:241–269, 1993.

[57] K. Veselić and I. Slapničar. Floating point perturbations of Hermitian matrices. *Lin. Alg. Appl.*, 195:81–116, 1993.

[58] P.-Å. Wedin. Perturbation theory for pseudoinverses. *BIT*, 13:217–232, 1973.

[59] J. H. Wilkinson and C. Reinsch, editors. *Handbook for Automatic Computation, vol 2.: Linear Algebra.* Springer-Verlag, Heidelberg, 1971.