ON COMPUTING ACCURATE SINGULAR
VALUES AND EIGENVALUES
OF ACYCLIC MATRICES

J.W. Demmel
W.B. Gragg

August 1992

Approved for public release; distribution unlimited
Prepared for: Naval Postgraduate School and
IFMAIA, Univ. of Minnesota,
Minneapolis, Minnesota 55455
This report was prepared in conjunction with research conducted for the Naval Postgraduate School and for the Institute for Mathematics and Its Applications. Funding was provided by the Naval Postgraduate School. Reproduction of all or part of this report is authorized.

Prepared by:

WILLIAM GRAGG
Professor of Mathematics

Reviewed by:

RICHARD FRANKE
Chairman
Department of Mathematics

Released by:

PAUL J. MARTO
Dean of Research
It is known that small relative perturbations in the entries of a bidiagonal matrix only cause small relative perturbations in its singular values, independent of the values of the matrix entries. In this paper we show that a matrix has this property if and only if its associated bipartite graph is acyclic. We also show how to compute the singular values of such a matrix to high relative accuracy. The same algorithm can compute eigenvalues of symmetric acyclic matrices with tiny componentwise relative backward error. This class includes tridiagonal matrices, arrow matrices, and exponentially many others.
On computing accurate singular values and eigenvalues of acyclic matrices

James W. Demmel *
Computer Science Division and Department of Mathematics
University of California
Berkeley, California 94720

William Gragg†
Department of Mathematics
Naval Postgraduate School
Monterey, California 93943

Abstract

It is known that small relative perturbations in the entries of a bidiagonal matrix only cause small relative perturbations in its singular values, independent of the values of the matrix entries. In this paper we show that a matrix has this property if and only if its associated bipartite graph is acyclic. We also show how to compute the singular values of such a matrix to high relative accuracy. The same algorithm can compute eigenvalues of symmetric acyclic matrices with tiny componentwise relative backward error. This class includes tridiagonal matrices, arrow matrices, and exponentially many others.

1 Introduction

In [9] it was shown that small relative perturbations in the entries of a bidiagonal matrix $B$ only cause small relative perturbations in its singular values. This is true independent of the values of the nonzero entries of $B$. This property justifies trying to compute the singular values of $B$ to high relative accuracy, and is essential to the error analyses of the corresponding algorithms [9].

Since this attractive property of bidiagonal matrices is independent of the values of the nonzero entries, it is really just a function of the sparsity pattern of bidiagonal matrices. In this paper we completely characterize those sparsity patterns with the property that independent of the values of the nonzero entries, small relative perturbations of the matrix entries only cause small relative perturbations of the singular values. The characterization

---

*The author was supported by NSF grant ASC-9005933 and DARPA grant DAAL03-91-C-0047 via a subcontract from the University of Tennessee. This work was performed during a visit to the Institute for Mathematics and its Applications at the University of Minnesota.

†The author also acknowledges the Institute for Mathematics and its Applications at the University of Minnesota.
is simple: a sparsity pattern has this property if and only if its associated bipartite graph is acyclic.

We define this graph as follows. Let $S$ be a sparsity pattern for $m$ by $n$ matrices; in other words, $S$ is a list of the entries permitted to be nonzero. Let $G(S)$ be a bipartite graph with one group of nodes $\{r_1, \ldots, r_m\}$ representing the $m$ rows and one group $\{c_1, \ldots, c_n\}$ representing the $n$ columns. There is an edge from $r_i$ to $c_j$ if and only if $A_{ij}$ is permitted to be nonzero. (We will sometimes write $G(A)$ instead of $G(S)$, where $S$ is the sparsity pattern of $A$.)

We also present another perturbation property of acyclic matrices which is quite strong: multiplying any single matrix entry by any factor $\beta \neq 0$ cannot change any singular value by more than a factor of $\beta$ (either up or down).

Sparsity patterns with this property have at most $n + m - 1$ nonzero entries. There are a great many such sparsity patterns. Let us consider only $m$ by $n$ sparsity patterns $S$ which cannot be permuted into block diagonal form (this means $G(S)$ is connected). Then the number of different such sparsity patterns is equal to the number of spanning trees on connected bipartite graphs with $m + n$ vertices; this number is $m^{n-1}n^{m-1}$ [5, p. 38] [3]. If we only wish to count sparsity patterns which cannot be made identical by reordering the rows and columns, a very simple lower bound on the number of such equivalence classes is $m^{n-1}n^{m-1}/(m!n!)$. In the square case $n = m$, Stirling's formula lets us approximate this lower bound by $\frac{\epsilon^{2n}}{(2\pi n^3)}$, which grows quickly.

Since we know the singular values of these acyclic matrices are determined to high relative accuracy by the data, it makes sense to try to compute them this accurately. We present a bisection algorithm which does this. The same algorithm can compute the eigenvalues of arbitrary "symmetric acyclic" matrices with tiny componentwise relative accuracy. We define symmetric acyclicity of a symmetric matrix as follows. Given a sparsity pattern $S$ of an $n$ by $n$ symmetric matrix, we define a graph $G'(S)$ by taking $n$ nodes, and connecting node $i$ to node $j \neq i$ if and only if the $(i,j)$ entry is nonzero. The symmetric sparsity pattern $S$ is called "symmetric acyclic" if the graph $G'(S)$ is acyclic. (We will sometimes write $G'(A)$ instead of $G'(S)$ where $S$ is the sparsity pattern of $A$.) The algorithm evaluates the inertia of such a matrix by doing symmetric Gaussian elimination, with the order of elimination determined by a postorder traversal of $G'(S)$.

In summary, the well-known attractive properties of bidiagonal matrices $B$ and symmetric tridiagonal matrices $T$, that the singular values of $B$ can be computed to high relative accuracy and the eigenvalues of $T$ computed with tiny componentwise relative backward error, have been extended to "acyclic" matrices. In the case of computing singular values, we have shown that this extension is complete: no other sparsity patterns have this property. We strongly suspect that the set of symmetric acyclic matrices is also the complete set of symmetric matrices whose eigenvalues can be computed with tiny componentwise relative backward error independent of the values of the matrix entries.

Other algorithms for the special case of "arrow" matrices are discussed in [1,2,15,22]. This work generalizes the adaptations of bisection to arrow matrices, and is almost certainly more stable than the QR based schemes.

The rest of this paper is organized as follows. Section 2 states the perturbation theorem for the singular values of acyclic matrices, and section 3 proves it. Section 4 shows how to compute eigenvalues of symmetric acyclic matrices with tiny componentwise relative
backward error, and applies this to compute the singular values of acyclic matrices to high relative accuracy. Section 5 gives some examples of matrices with acyclic sparsity patterns. Section 6 discusses algorithms and open problems.

2 Statement of Perturbation Theorem for Singular Values

In this section we define three properties of sparsity patterns of matrices, one about graph theory and two about perturbation theory. Our main result, which we prove in the next section, is that these properties are equivalent.

Let $A$ be an $m \times n$ matrix with a fixed sparsity pattern $S$.

**Property 1.** $G(S)$ is acyclic.

**Property 2.** Given sparsity pattern $S$, there exist positive constants $\epsilon_0$ and $\zeta$ with the following property. Let $A$ be any matrix with sparsity pattern $S$, and $A_{ij}$ any nonzero entry. Choose any $|\epsilon| \leq \epsilon_0$, and let $A' = A$ except for $A'_{ij} = A_{ij}(1 + \epsilon)$. Then for all singular values $\sigma_k(A')$

$$(1 - \zeta|\epsilon|)\sigma_k(A) \leq \sigma_k(A') \leq (1 + \zeta|\epsilon|)\sigma_k(A)$$

In other words a sufficiently small relative perturbation $\epsilon$ in any single matrix entry cannot cause a relative perturbation greater than $\zeta \epsilon$ in any singular value.

If $p$ entries of $A$ are simultaneously perturbed, Property 2 can be applied $p$ times to show no singular value can change by a factor outside the interval from $(1 - \zeta|\epsilon|)^p = 1 - p\zeta|\epsilon| - O(\epsilon^2)$ to $(1 + \zeta|\epsilon|)^p = 1 + p\zeta|\epsilon| - O(\epsilon^2)$. Property 2 seems rather weak, since it imposes no bounds on $\epsilon_0$ nor $\zeta$. Still, since $\epsilon_0$ and $\zeta$ are independent of the matrix entries, it is actually demanding quite a bit of $S$. The last property is even stronger:

**Property 3.** Given sparsity pattern $S$, let $A$ be any matrix with this sparsity, and $A_{ij}$ any nonzero entry. Let $\beta$ be any nonzero constant. Let $A' = A$ except for $A'_{ij} = \beta A_{ij}$. Then for all singular values $\sigma_k(A')$

$$\min(|\beta|, |\beta^{-1}|)\sigma_k(A) \leq \sigma_k(A') \leq \max(|\beta|, |\beta^{-1}|)\sigma_k(A)$$

Property 3 is much stronger than Property 2 because it imposes no limit $\epsilon_0$ on the size of the relative perturbation, and because it asserts $\zeta = 1$, i.e. that the relative change in the singular values cannot exceed the relative change in the single perturbed matrix entry. In the case of simultaneous small relative perturbations of size at most $\beta = 1 + \epsilon$ in $p$ entries of $A$. Property 3 implies that no singular value can change by a factor outside the interval from $(1 - |\epsilon|)^p = 1 - p|\epsilon| + (\epsilon^2)$ to $(1 - |\epsilon|)^{-p} = 1 + p|\epsilon| + (\epsilon^2)$. Since the maximum number of nonzeros is $m + n - 1$, this relative perturbation is bounded by $(m + n - 1)|\epsilon| + O(\epsilon^2)$.

Our main result is

**Theorem 1.** Properties 1, 2 and 3 of a sparsity pattern $S$ are equivalent.
Figure 1: Computing $D_r$ and $D_c$

if $q$ is a row node then
  suppose $q = r_i$
  if $r_i$ is the root then
    $D_{r,ii} = 1$
  else
    suppose $c_j$ is the parent of $q$
    $D_{r,ii} = 1/(A_{ij} D_{c,jj})$
  end
else (q must be a column node) then
  suppose $q = c_j$
  if $c_j$ is the root then
    $D_{c,jj} = 1$
  else
    suppose $r_i$ is the parent of $q$
    $D_{c,jj} = 1/(A_{ij} D_{r,ii})$
  end
end if

3 Proof of Perturbation Theorem for Singular Values

The proof of equivalence will consist of the following steps. We already know that Property 3 implies Property 2, so it will suffice to prove Property 1 implies Property 3, and Property 2 implies Property 1.

Lemma 1 Let $A$ have sparsity pattern $S$, and suppose $G(S)$ is acyclic. Then there are diagonal matrices $D_r$ and $D_c$ such that each entry of $D_r A D_c$ is either 0 or 1. Each diagonal entry of $D_r$ or $D_c$ is a quotient of monomials in the entries of $A$. In each monomial each distinct factor $A_{ij}$ which appears has unit exponent. Each $A_{ij}$ can appear only in numerators of entries of $D_r$ and denominators of entries of $D_c$, or vice versa, in denominators of entries of $D_r$ and numerators of entries of $D_c$.

Proof Since $G(S)$ is acyclic, it is a forest of trees. We may consider each tree independently. We traverse each tree via depth first search, and execute the program in Figure 1 when first visiting node $q$.

The depth first search visits each node once. Since the graph is bipartite, row nodes and column nodes alternate, so the parent of a row node is a column node and vice versa. Since each node is visited once, the above program is executed once for each edge in the tree, i.e. once for each nonzero entry $A_{ij}$, corresponding to the edge connecting nodes $r_i$ and $c_j$. Thus each $D_{r,ii}$ and $D_{c,jj}$ is set exactly once. Since the $i,j$ entry of $D_r A D_c$ is $D_{r,ii} A_{ij} D_{c,jj}$, we see immediately from the way $D_{r,ii}$ and $D_{c,jj}$ are defined that this quantity is 1 if $A_{ij} \neq 0$ (and 0 otherwise). Since each $A_{ij}$ is used once during the graph traversal, each $D_{r,ii}$ and
must be a quotient of monomials. If \( A_{ij} \) is first used in \( D_{r,ii} \), then the formulas in the above program and the fact the row and column nodes alternate mean that \( A_{ij} \) will only appear in denominators of entries of \( D_r \) and numerators of entries of \( D_c \). Alternatively, if \( A_{ij} \) is first used in \( D_{c,ij} \), then \( A_{ij} \) will only appear in denominators of entries of \( D_c \) and numerators of entries of \( D_r \). □

The rest of the proof mimics that of [4, Thm. 1]. Let \( E \) be the matrix of ones and zeros with sparsity \( S \), so that \( D_r A D_c = E \). Write \( D_c = S_c |D_r| \) where \( |D_r| \) is the matrix of absolute values of \( D_r \), and \( S_r \) is a diagonal matrix with \( |S_r| = I \). Similarly write \( D_c = S_c |D_c| \). Then

\[
A = D_r^{-1} E D_c^{-1} = S_r^{-1} |D_r|^{-1} E |D_c|^{-1} S_c^{-1} = S_c^{-1} |A| S_c^{-1}
\]

so that \( A \) is related to \(|A|\) by pre- and postmultiplication by diagonal orthogonal matrices. In particular, \( A \) and \(|A|\) have the same singular values. We will henceforth assume without loss of generality that \( A \) is nonnegative and so \( D_r \) and \( D_c \) are also nonnegative.

It is known that the singular values of \( A \) are the same as the positive eigenvalues of the pencil

\[
\begin{bmatrix}
0 & A \\
A^T & 0
\end{bmatrix} - \lambda I
\]

which are in turn the same as the positive eigenvalues of the equivalent symmetric definite pencil

\[
\begin{bmatrix}
D_r & 0 \\
0 & D_c
\end{bmatrix} \begin{bmatrix}
0 & A \\
A^T & 0
\end{bmatrix} - \lambda \cdot I
\begin{bmatrix}
0 & E \\
E^T & 0
\end{bmatrix} - \lambda \cdot \begin{bmatrix}
D_r^2 & 0 \\
0 & D_c^2
\end{bmatrix} \equiv F - \lambda D^2
\]

Now suppose we perturb \( A \) by changing nonzero entry \( A_{ij} \) to \( \beta A_{ij} \), resulting in the perturbed matrix \( A' \). Apply the algorithm in Lemma 1 to compute a new \( D'_r \) and \( D'_c \). Since by Lemma 1 the entries of \( D'_r \) and \( D'_c \) are quotients of monomials where each independent factor appears at most once, each entry \( D'_{r,kk} \) must equal either \( D_{r,kk} \), \( \beta D_{r,kk} \) or \( \beta^{-1} D_{r,kk} \). An analogous statement about \( D'_{c,kk} \) and \( D_{c,kk} \) is true. Since a factor \( A_{ij} \) must appear either in numerators of \( D_r \) and denominators of \( D_c \), or in denominators of \( D_c \) and numerators of \( D_r \), we have two cases:

1. Either \( D'_{r,kk} = D_{r,kk} \) or \( D'_{r,kk} = \beta D_{r,kk} \), and either \( D'_{c,kk} = D_{c,kk} \) or \( D'_{c,kk} = \beta^{-1} D_{c,kk} \).

2. Either \( D'_{r,kk} = \beta D_{r,kk} \) or \( D'_{r,kk} = \beta^{-1} D_{r,kk} \), and either \( D'_{c,kk} = D_{c,kk} \) or \( D'_{c,kk} = \beta D_{c,kk} \).

Note we may multiply \( D_r \) by any nonzero \( \gamma \) and divide \( D_c \) by \( \gamma \) without changing the fact that \( D_r A D_c = E \). Corresponding to the above two cases, we

1. divide \( D_r \) by \(|\beta|^{1/2} \) and multiply \( D_c \) by \(|\beta|^{1/2} \), or

2. divide \( D_c \) by \(|\beta|^{1/2} \) and multiply \( D_r \) by \(|\beta|^{1/2} \).

The end result will be \( D'_r \) and \( D'_c \) matrices each of whose entries differs from the corresponding entry of \( D_r \) and \( D_c \) by factors of \(|\beta|^{\pm 1/2} \). In particular, this implies

\[
|\beta|^{-1} \leq \frac{x^T D_r^2 x}{x^T D'_r x} \leq |\beta| \quad \text{and} \quad |\beta|^{-1} \leq \frac{x^T D_c^2 x}{x^T D'_c x} \leq |\beta|
\]

\[
\text{5}
\]
for any nonzero vector \( x \). Let \( D' = \text{diag}(D'_1, D'_2) \) as we above defined \( D = \text{diag}(D_1, D_2) \).

Then

\[
|\beta|^{-1} \leq \frac{y^T D^2 y}{y^T y} \leq |\beta|
\]

for any nonzero vector \( y \). We may now apply [4, Lemma 2] to conclude that

\[
\sigma_k(A) = \min_{S^k} \max_{x \in S^k} \frac{x^T F z}{x^T D^2 x},
\]

\[
\sigma_k(A') = \min_{S^k} \max_{x \in S^k} \frac{x^T F z}{x^T D'^2 x},
\]

where the minima are over all \( k \) dimensional subspaces \( S^k \), can differ by no more than a factor of \( \beta \). This proves that Property 1 implies Property 3.

**Lemma 2** Let \( A \) have sparsity pattern \( S \), and let all its nonzero entries be independent indeterminates. Then \( G(S) \) is acyclic if and only if all minors of \( A \) are either 0 or monomials.

Proof: We begin by noting that to each term in the determinant of an \( s \)-by-\( s \) square matrix \( M \) corresponds a unique perfect matching in graph \( G(M) \). This is because each term in the determinant corresponds to a choice of \( s \) entries of \( M \) located in disjoint rows and columns, and each such choice of \( s \) entries selects a perfect match in \( G(M) \).

Now suppose a square submatrix \( M \) of \( A \) has at least two terms in its determinant. These correspond to two different perfect matchings. Take the symmetric difference of the edges in these matchings. This symmetric difference forms a cycle, which we get by following edges of the two matchings in alternation. Thus \( G(M) \) contains a cycle, and so must \( G(A) \) since it includes \( G(M) \).

Now suppose \( G(A) \) contains a cycle. Assume without loss of generality that it is a simple cycle, i.e., it is connected and visits each node once. Let \( M \) be the corresponding square submatrix. This cycle determines two perfect matchings in \( G(M) \), consisting of alternate edges of the cycle. This means \( \text{det}(M) \) has at least two terms. \( \Box \)

To prove that Property 2 implies Property 1, we will show the contrapositive. So assume \( G(A) \) contains a cycle, and let \( M \) be an \( s \)-by-\( s \) submatrix whose determinant has at least 2 terms. This means we may choose all the entries of \( M \) to be nonzero but such that \( M \) is exactly singular. Thus its singular values include at least one which is exactly zero. Scale \( M \) so that its entry of smallest absolute value is 1, and let \( \sigma = \|M\|_2 \geq 1 \). Now let \( A(M, \eta) \) denote the matrix with sparsity \( S \), submatrix \( M \), and other nonzero entries equal to \( \eta \). Then \( A(M, \eta) \) will have at least \( \min(m, n) - s + 1 \) zero singular values, \( \min(m, n) - s \) from the zero rows and columns outside \( M \), and 1 from the singularity of \( M \). By standard perturbation theory \( A(M, \eta) \) will have at least \( \min(m, n) - s + 1 \) singular values no larger than \( m\eta \). Now change a smallest entry of \( M \) from 1 to \( 1 + \epsilon \) to get \( M_\epsilon \); thus \( \epsilon \) is also the
relative change in this entry. Then \(|\det(M_x)| \geq x\), and so \(\sigma_{\text{min}}(M_x) \geq |x|/(\sigma + x)^{s-1}\). This means \(\sigma_s(A(M, \eta)) \geq |x|/(\sigma + x)^{s+1} - mn\eta\), whereas \(\sigma_s(A(M, \eta)) \leq mn\eta\). Thus
\[
\frac{\sigma_s(A(M_x, \eta))}{\sigma_s(A(M, \eta))} \geq \frac{x}{(\sigma + x)^{s+1} - mn\eta} = \frac{x}{mn\eta(\sigma + x)^{s+1}} - 1.
\]

If Property 1 held, then we would be able to find \(\epsilon_0 > 0\) and \(\zeta > 0\) such that for all \(0 < x < \epsilon_0\) and \(\eta > 0\) the following inequality would hold:
\[
\frac{x}{mn\eta(\sigma + x)^{s+1}} - 1 \leq \zeta.
\]

Since we can make \(\eta\) as small as we like, this inequality cannot hold for any finite \(\zeta\). Thus Property 2 cannot hold. This completes the proof that Property 2 implies Property 1, and so also completes the proof of Theorem 1.

4 A bisection algorithm for computing eigenvalues with tiny backward error

Let \(\varepsilon_M\) denote the machine precision. We will assume the usual model of floating point error: \(\text{fl}(a - b) = (a \oplus b)(1 + \delta)\) with \(|\delta| \leq \varepsilon_M\), and assume neither underflow nor overflow occur. (Of course, a practical algorithm would need to account for overflow. This can be done analogously to the way overflow is accounted for in standard tridiagonal bisection [13].)

In this section we will show how to compute the eigenvalues of a symmetric acyclic matrix \(T\) with tiny componentwise relative backward error. Our main result is

**Theorem 2** The algorithm in Figure 2 computes \(\text{count}(T, x)\), the number of eigenvalues of \(T\) less than \(x\), with a backward error \(\delta T\) with the following properties:

\[
|\delta T_{ij}| \leq (1.5v + 2.5)\varepsilon_M |T_{ij}| \quad \text{when} \quad i \neq j.
\]
\[
|\delta T_{ii}| \leq (2v + 2)\varepsilon_M |x|.
\]

Here \(v \leq n - 1\) is the maximum degree of any node in the graph of \(T\). In other words, the computed \(\text{count}(T, x)\) is the exact value of \(\text{count}(T + \delta T, x)\) where \(\delta T\) is bounded as above.

This is essentially identical to the standard error analysis of Sturm sequence evaluation for symmetric tridiagonal matrices [9, Sec. 6] [13] (this is stronger than the result in [20, p. 303]).

Our algorithm simply performs symmetric Gaussian elimination on \(T - xI\): \(P(T - xI)P^T = LDL^T\) where \(P\) is a permutation matrix, \(L\) is unit lower triangular and \(D\) is diagonal. Then \(\text{count}(T, x)\) is simply the number of negative diagonal entries of \(D\), by Sylvester's Inertia Theorem [16]. The order of elimination is the same as a postorder traversal of the nodes of the acyclic graph. Since leaves, which have degree 1, are eliminated first, there is no fill-in during the elimination, and all off-diagonal entries \(L_{ij}\) of \(L\) can be computed by simply dividing \(L_{ij} = T_{ij}/D_{jj}\).
Figure 2: Computing \( \text{count}(T, x) \)

call \( \text{cnt}(i, d, s, x) \) where \( i \) is any node \( 1 \leq i \leq n \)
return \( \text{count}(T, x) = s \)

procedure \( \text{cnt}(i, d, s, x) \)
/* \( i \) and \( x \) are input parameters, \( d \) and \( s \) are output parameters */
\( d = T_{ii} - x \)
\( s = 0 \)
for all children \( j \) of \( i \) do
  call \( \text{cnt}(j, d', s', x) \)
  \( d = d - T_{ij}^2 / d' \)
  \( s = s + s' \)
end for
if \( d < 0 \), then \( s = s + 1 \)
return \( d \) and \( s \)
end procedure

We assume the graph \( G'(S) \) is connected, since otherwise the matrix can be reordered to be block diagonal (one diagonal block per connected component of \( G'(S) \)), and the inertia of each diagonal block can be computed separately. The algorithm \( \text{cnt}(i, d, s, x) \) in Figure 2 assumes the matrix is stored in graph form. Subroutine \( \text{cnt}(i, d, s, x) \) does a postorder traversal of the acyclic graph \( G'(S) \), and may be called starting at any node \( 1 \leq i \leq n \). In addition to \( i, x \) is an input parameter. The variables \( d \) and \( s \) are output parameters; on return \( s \) is the desired value of \( \text{count}(T, x) \).

To prove Theorem 2, we will exploit the acyclicity of \( T \) to show that each computed quantity and original entry of \( T \) is used (directly) just once during the entire computation, and then use this to "push" the rounding error back to the original data.

We see that each entry of \( T \) is used just once as follows. \( T_{ii} \) is only used when visiting node \( i \), and \( T_{ij} \) is used only once, when visiting \( i \) if \( j \) is a child of \( i \) or when visiting \( j \) if \( i \) is a child of \( j \) in the postorder traversal tree.

Now denote the \( d \) computed when visiting node \( i \) by \( d_i \). The floating point operations performed while visiting node \( i \) are then

\[
 d_i = fl \left( T_{ii} - x - \sum_{\text{all children}} \frac{T_{ij}^2}{d_j} \right) \tag{4.1}
\]

To analyze this formula, we will let subscripted \( \epsilon \)'s denote independent quantities bounded in absolute value by \( \epsilon_M \). We will also make standard approximations like

\[(1 + \epsilon_1)^{\pm 1}(1 + \epsilon_2)^{\pm 1} = 1 + 2\epsilon_3.\]
Since we do not know the number of terms or the order of the sum in equation (4.1), we will make the worst case assumption that there are \( v < n-1 \) terms where \( v \) is the maximum degree of any node in the graph \( G'(S) \). This leads to

\[
d_i = (1 + (v + 1)\varepsilon_{ia})T_{ii} - (1 + (v + 1)\varepsilon_{ia})x - \sum_{\text{all children} \ j \ of \ i} (1 + (v + 3)\varepsilon_{ij}) \frac{T_{ij}^2}{d_j} \tag{4.2}
\]

or

\[
d_i = \frac{T_{ii} - x + (2v + 2)\varepsilon_{ic}x - \sum_{\text{all children} \ j \ of \ i} \frac{((1 + (v + 2)\varepsilon_{ij})T_{ij})^2}{d_j}}{1 + (v + 1)\varepsilon_{ia}} \tag{4.3}
\]

Let \( \varepsilon_{ia} \) be the roundoff error corresponding to \( \varepsilon_{ia} \) committed when computing \( d_j \). Then

\[
d_i = \frac{T_{ii} - x + (2v + 2)\varepsilon_{ic}x - \sum_{\text{all children} \ j \ of \ i} \frac{((1 + (1.5v + 2.5)\varepsilon_{ij})T_{ij})^2}{d_j/(1 + (v + 1)\varepsilon_{ja})}}{1 + (v + 1)\varepsilon_{ia}} \tag{4.4}
\]

or, finally,

\[
d'_i = T_{ii} - x + (2v + 2)\varepsilon_{ic}x - \sum_{\text{all children} \ j \ of \ i} \frac{((1 + (1.5v + 2.5)\varepsilon_{ij})T_{ij})^2}{d'_j} \tag{4.5}
\]

where \( d'_i = d_i/(1 + \varepsilon_{ia}) \). Equation (4.5) tells us that the \( d'_i \) are the exact diagonal entries of \( D \) in \( P(T + \delta T - \mu I)P^T = LDL^T \). Since they obviously have the same signs as the \( d_i \), this proves Theorem 2.

The proof depends strongly on there not being any fill-in and on each off diagonal entry being computable by a single division. Since these properties hold if and only if the graph \( G'(T) \) is symmetric acyclic, we strongly suspect that this is the only class of matrices whose eigenvalues can always be computed with tiny componentwise relative backward error.

We now apply Theorem 2 to compute singular values of acyclic matrices to high relative accuracy. So suppose \( B \) is a matrix whose graph \( G(B) \) is acyclic. Consider the symmetric matrix

\[
A = \begin{bmatrix} 0 & B \\ B^T & 0 \end{bmatrix}
\]

It is well known that the positive eigenvalues of \( A \) are the singular values of \( B \). It is also immediate that the graph \( G'(A) = G(B) \). Therefore \( B \) is acyclic if and only if \( A \) is symmetric acyclic, so we can apply the above algorithm to compute all \( B \)'s singular values to high relative accuracy.

One other algorithm is worth mentioning. If \( A \) is symmetric positive definite and symmetric acyclic, then its Cholesky factor \( L \) is acyclic, has the "lower half" of the sparsity
pattern of \( A \) and may be computed by using algorithm cnt. It may occasionally be more accurate to compute \( A \)'s eigenvalues by first computing \( L \), computing its singular values by bisection, and then squaring the singular values to get \( A \)'s eigenvalues [4]. This is the case, for example, for the tridiagonal matrix with 2's on the diagonal and 1's on the off-diagonal.

5 Examples

We give various examples of acyclic sparsity patterns, beginning with acyclic \( G(S) \). Given any acyclic sparsity pattern, others can be generated either by permuting rows and/or columns, or by adding more zeros. Since all square acyclic matrices have monomial (or zero) determinants, this means we can permute them to be upper triangular. In addition to bidiagonal matrices, some other examples are

\[
\begin{bmatrix}
  x & x \\
  x & x \\
  x & x \\
  x & x \\
\end{bmatrix}
\]

and

\[
\begin{bmatrix}
  x & x \\
  x & x \\
  x & x \\
  x & x \\
\end{bmatrix}
\]

To get symmetric acyclic matrices \( A \), one can always take an acyclic \( B \) and set

\[
A = \left[ \begin{array}{cc}
0 & B \\
B^T & 0
\end{array} \right] - \lambda I.
\]

Some other examples are

\[
\begin{bmatrix}
  x & x \\
  x & x \\
  x & x \\
  x & x \\
\end{bmatrix}
\]

and

\[
\begin{bmatrix}
  x & x \\
  x & x \\
  x & x \\
  x & x \\
\end{bmatrix}
\]

6 Algorithms and Open Problems

In [8] a perturbation theorem for singular vectors of bidiagonal matrices is proven, which shows that the appropriate condition number for the \( i \)-th singular vector is the reciprocal of the relative difference between the \( i \)-th singular value and next closest one. It would be interesting to extend this to the acyclic case.

Given the perturbation theory, it would be nice to compute the singular vectors as accurately as they deserve. A natural candidate is inverse iteration, but even in the simple case of symmetric tridiagonal matrices, open problems remain. In particular there is no absolute guarantee that the computed eigenvectors are orthogonal, although in practice the algorithm can be made quite robust [11].

In the “extreme” cases of tridiagonal and arrow matrices, we know how to compute the inertia in \( O(\log n) \) time, using the so-called parallel-prefix algorithm in the tridiagonal case [17,19] and more simply in the arrow case. The stability in the tridiagonal case is unknown,
but in practice it appears to be stable. We can extend this to the general symmetric acyclic case in two ways. First, the tree describing the expression whose final value is $d_i$ has at most $n$ leaves. From [6] we know any such expression tree can be evaluated in at most $4 \log_2 n$ parallel steps, although stability may be lost. Another approach, which includes parallel prefix and the algorithm in [15] as special cases, is based on [14]. The idea is to simply evaluate the tree greedily, summing $k$ leaves of a single node in $O(\log_2 k)$ steps whenever possible, and collapsing a chain of $k$ nodes into a single node via parallel prefix in $O(\log_2 k)$ steps whenever possible. If we could understand the numerical stability of parallel prefix, we could probably analyze this more general scheme as well.

Divide and conquer [7,10,18,12] has been widely used for the tridiagonal eigenproblem and bidiagonal singular value decomposition. This can be straightforwardly extended to the acyclic case. In terms of the tree, just remove the root by a "rank one tearing", solve the independent child subtrees recursively and in parallel, and merge the results by solving the secular equation [21]. Any node can be the root, and to be efficient it is important that no subtree be large. In the tridiagonal case, there are always two subtrees of nearly equal size. In a general tree one can only make sure that no subtree has more than half the nodes of the original tree (this is easily done in $O(n)$ time via depth first search).

QR does not appear to extend beyond the tridiagonal case. The case of arrow matrices was analyzed in [2], where it was shown that no QR algorithm could exist. A simpler proof arises from noting that two steps of $LL^T$ is equivalent to one step of QR in the positive definite case, and so the question is whether the sparsity pattern of $T_0 = LL^T$ is the same as that of $T_1 = L^T L$; this is easily seen to include only tridiagonal $T_0$.

Acknowledgement

The authors thank Richard Brualdi, who pointed out references [3,5].

References


[8] P. Deift, J. Demmel, L.-C. Li, and C. Tomei. The bidiagonal singular values decom-


[10] J. Dongarra and D. Sorensen. A fully parallel algorithm for the symmetric eigenprob-


[12] E. Jessup and D. Sorensen. A parallel algorithm for computing the singular value decom-
position of a matrix. Mathematics and Computer Science Division Report ANL/MCS-
TM-102, Argonne National Laboratory, Argonne, IL, December 1987.

Dept. Technical Report CS41, Stanford University, Stanford, CA, July 1966 (revised
June 1968).

IEEE. 1985.

[15] D. O'Leary and G. W. Stewart. Computing the eigenvalues and eigenvectors of sym-

1980.


[18] D. Sorensen and P. Tang. On the orthogonality of eigenvectors computed by divide-

[19] P. Swarztrauber. A parallel algorithm for computing the eigenvalues of a symmetric

1965.

[21] Zhonggang Zeng. The acyclic eigenproblem can be reduced to the arrowhead one.
preprint, Michigan State University, 1992.

[22] Hongyuan Zha. A two-way chasing scheme for reducing a symmetric arrowhead matrix
to tridiagonal form. Scientific computing and computational mathematics, Stanford
University, 1992.
DISTRIBUTION LIST

DIRECTOR (2)
DEFENSE TECH. INFORMATION CENTER, CAMERON STATION
ALEXANDRIA, VA 22314

DIRECTOR OF RESEARCH ADMIN.
CODE 81
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CA 93943

LIBRARY (2)
CODE 52
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CA 93943

NATIONAL SCIENCE FOUNDATION
WASHINGTON, D.C. 20550

DEPARTMENT OF MATHEMATICS
CODE MA
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CA 93943

PROFESSOR WILLIAM GRAGG (20)
CODE MA/GR
DEPARTMENT OF MATHEMATICS
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CA 93943