Bounding the error in Gaussian elimination for tridiagonal systems

Higham, Nicholas J.

1990

MIMS EPrint: **2006.172**

Manchester Institute for Mathematical Sciences
School of Mathematics

The University of Manchester

Reports available from: http://eprints.maths.manchester.ac.uk/
And by contacting: The MIMS Secretary
School of Mathematics
The University of Manchester
Manchester, M13 9PL, UK

ISSN 1749-9097

SIAM J. MATRIX ANAL. APPL.
Vol. 11, No. 4, pp. 521-530, October 1990

003

BOUNDING THE ERROR IN GAUSSIAN ELIMINATION FOR TRIDIAGONAL SYSTEMS*

NICHOLAS J. HIGHAM†

Abstract. If $\hat{x}$ is the computed solution to a tridiagonal system $Ax = b$ obtained by Gaussian elimination, what is the "best" bound available for the error $x - \hat{x}$ and how can it be computed efficiently? This question is answered using backward error analysis, perturbation theory, and properties of the LU factorization of A. For three practically important classes of tridiagonal matrix, those that are symmetric positive definite, totally nonnegative, or M-matrices, it is shown that $(A + E)\hat{x} = b$ where the backward error matrix E is small componentwise relative to A. For these classes of matrices the appropriate forward error bound involves Skeel's condition number cond (A, x), which, it is shown, can be computed exactly in $O(n)$ operations. For diagonally dominant tridiagonal A the same type of backward error result holds, and the author obtains a useful upper bound for cond (A, x) that can be computed in $O(n)$ operations. Error bounds and their computation for general tridiagonal matrices are discussed also.

Key words. tridiagonal matrix, forward error analysis, backward error analysis, condition number, comparison matrix, M-matrix, totally nonnegative, positive definite, diagonally dominant, LAPACK

AMS(MOS) subject classifications. primary 65F05, 65G05

C.R. classification. G.1.3

1. Introduction.

A natural question to ask when solving a general $n \times n$ linear system $Ax = b$ by Gaussian elimination with partial pivoting (GEPP) is, "how accurate is the computed solution, $\hat{x}$?" The traditional answer begins with Wilkinson's backward error result [22, p. 108]

$$(A+F)\hat{x}=b, \qquad \|F\|_\infty \leqq \rho_n p(n) u \|A\|_\infty, \tag{1.1}$$

where $p(n)$ is a cubic polynomial, u is the unit roundoff, and ρ_n is the *growth factor*, defined in terms of the quantities $a_{ij}^{(k)}$ generated during the elimination by

$$\rho_n = \frac{\max_{i,j,k} |a_{ij}^{(k)}|}{\max_{i,j} |a_{ij}|}.$$

Applying standard perturbation theory to (1.1), one obtains the forward error bound

$$\frac{\|x-\hat{x}\|_\infty}{\|x\|_\infty} \leqq \frac{\kappa_\infty(A)\rho_n p(n) u}{1-\kappa_\infty(A)\rho_n p(n) u} \qquad (\kappa_\infty(A)\rho_n p(n) u < 1), \tag{1.2}$$

where the condition number $\kappa_\infty(A) = \|A\|_\infty \|A^{-1}\|_\infty$. Since the term $p(n)$ can usually be replaced by its square root for practical purposes [22, p. 108], or more crudely can be ignored, and since ρ_n is usually of order 1, this leads to the rule of thumb that $\hat{x}$ has about $-\log_{10} u - \log_{10} \kappa_\infty(A)$ correct decimal digits in its largest component.

In certain circumstances a bound potentially much smaller than (1.2) holds. This can be shown using the following componentwise backward error result, for general A [5]:

$$(A+E)\hat{x}=b, \qquad |E| \leqq c_n u |\hat{L}|\,|\hat{U}|, \tag{1.3}$$

* Received by the editors February 13, 1989; accepted for publication (in revised form) September 1, 1989.

† Department of Computer Science, Upson Hall, Cornell University, Ithaca, New York 14853 (na.nhigham@na-net.stanford.edu). Present address, Department of Mathematics, University of Manchester, Manchester, M13 9PL, United Kingdom.

where $c_n = 2n + O(u)$, and $\hat{L}$ and $\hat{U}$ are the computed LU factors of A (we assume, without loss of generality, that there are no row interchanges). Here, the absolute value operation $|\cdot|$ and the matrix inequality are interpreted componentwise. If $|\hat{L}||\hat{U}| \leqq c_n'|A|$, then (1.3) may be written

$$(A+E)\hat{x} = b, \qquad |E| \leqq c_n'' u |A|, \tag{1.4}$$

which represents the "ideal" situation where E is small componentwise relative to A. Note, in particular, that $e_{ij} = 0$ if $a_{ij} = 0$. The bound in (1.4) holds, at least, when A is triangular (see, e.g., [17]), and when A is *totally nonnegative* [5], assuming no pivoting in both cases. (A is totally nonnegative if all its minors of any order are nonnegative.) The bound also holds, under certain assumptions, if $\hat{x}$ is the result of GEPP followed by one step of iterative refinement in single precision [1], [20].

Perturbation results appropriate to (1.4) render the bound [19]

$$\frac{\|x-\hat{x}\|_\infty}{\|x\|_\infty} \leqq \frac{\operatorname{cond}(A,x)c_n'' u}{1-\operatorname{cond}(A)c_n'' u} \qquad (\operatorname{cond}(A)c_n'' u < 1), \tag{1.5}$$

where

$$\operatorname{cond}(A,x) = \frac{\| |A^{-1}||A||x| \|_\infty}{\|x\|_\infty}$$

and

$$\operatorname{cond}(A) = \operatorname{cond}(A,e), \qquad e = (1, 1, \cdots, 1)^T.$$

The key difference between (1.5) and (1.2) is in the condition number terms: $\operatorname{cond}(A,x)$ is no larger than $\kappa_\infty(A)$ and is often much smaller. In particular, $\operatorname{cond}(A,x)$ is invariant under row scaling of A, whereas $\kappa_\infty(A)$ is not.

This work focuses on the case where A is tridiagonal, and was partly motivated by the question of what types of error bounds and condition number estimates should be provided in the LAPACK routines for solving tridiagonal systems [3], [9]. (LAPACK is to be a collection of Fortran 77 routines for solving linear equations, linear least squares problems, and matrix eigenvalue problems [6].) The aim of the work is to determine classes of tridiagonal systems for which the bounds (1.4) and (1.5) are valid and to develop efficient methods for estimating or computing the condition numbers in (1.5) and (1.2).

In § 2 we present a specialized version of the backward error bound (1.3) for tridiagonal matrices. The result is known, but we give a short proof since the precise value of the bound is important, and we were unable to find a suitable reference.

In § 3 we show that (1.4) holds for Gaussian elimination without pivoting if the tridiagonal matrix A is symmetric positive definite, totally nonnegative, or an M-matrix. (Thus, for these types of matrices there is no advantage in doing iterative refinement in single precision.) We show that in each case $\operatorname{cond}(A,x)$, and hence also the bound in (1.5), can be computed exactly in $O(n)$ operations. Diagonally dominant matrices also enjoy a relatively small componentwise backward error, and, as we show in § 4, a good upper bound for $\operatorname{cond}(A,x)$ can be obtained in $O(n)$ operations.

We consider general tridiagonal matrices in § 5; we explain which error bounds are applicable and how the corresponding condition numbers may be estimated. In § 6 some further comments are made concerning practical use of the bounds and condition numbers, and some numerical results are presented to illustrate the value of using a componentwise backward error approach when possible.

2. Gaussian elimination and its error analysis. Consider the real $n \times n$, nonsingular tridiagonal matrix

$$A = \begin{bmatrix} d_1 & e_1 & & & \\ c_2 & d_2 & e_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & e_{n-1} \\ & & & c_n & d_n \end{bmatrix}, \tag{2.1}$$

and assume A has an LU factorization $A = LU$, where

$$L = \begin{bmatrix} 1 & & & & \\ l_2 & 1 & & & \\ & l_3 & 1 & & \\ & & \ddots & \ddots & \\ & & & l_n & 1 \end{bmatrix}, \qquad U = \begin{bmatrix} u_1 & e_1 & & & \\ & u_2 & e_2 & & \\ & & \ddots & \ddots & \\ & & & \ddots & e_{n-1} \\ & & & & u_n \end{bmatrix}. \tag{2.2}$$

Gaussian elimination for computing L and U is described by the recurrence relations

$$u_1 = d_1; \qquad \left.\begin{aligned} l_i &= c_i/u_{i-1} \\ u_i &= d_i - l_i e_{i-1} \end{aligned}\right\} i = 2, \cdots, n. \tag{2.3}$$

To investigate the effects of rounding error, we will employ the model

$$fl(x \text{ op } y) = (x \text{ op } y)(1+\delta), \qquad |\delta| \leqq u, \tag{2.4a}$$

$$fl(x \text{ op } y) = (x \text{ op } y)/(1+\varepsilon), \qquad |\varepsilon| \leqq u, \tag{2.4b}$$

where u is the unit roundoff and $\text{op} \in \{+, -, *, /\}$. Note that (2.4b) is valid under the same assumptions as (2.4a), although usually only (2.4a) is used in a rounding error analysis. Judicious use of (2.4b) simplifies the analysis slightly.

Applying (2.4) to the relations (2.3) and using a hat to denote computed quantities, we have

$$(1+\varepsilon_i)\hat{l}_i = \frac{c_i}{\hat{u}_{i-1}}, \qquad |\varepsilon_i| \leqq u,$$

$$(1+\theta_i)\hat{u}_i = d_i - \hat{l}_i e_{i-1}(1+\delta_i), \qquad |\theta_i|, |\delta_i| \leqq u.$$

Hence

$$|c_i - \hat{l}_i \hat{u}_{i-1}| \leqq u|\hat{l}_i \hat{u}_{i-1}|,$$

$$|d_i - \hat{l}_i e_{i-1} - \hat{u}_i| \leqq u(|\hat{l}_i e_{i-1}| + |\hat{u}_i|).$$

In matrix terms these bounds may be written as

$$A = \hat{L}\hat{U} + E, \qquad |E| \leqq u|\hat{L}||\hat{U}|. \tag{2.5}$$

If the LU factorization is used to solve a system $Ax = b$ by forward and back substitution, then it is straightforward to show that the computed solution $\hat{x}$ satisfies

$$(\hat{L} + \Delta L)(\hat{U} + \Delta U)\hat{x} = b, \qquad |\Delta L| \leqq u|\hat{L}|, \qquad |\Delta U| \leqq (2u + u^2)|\hat{U}|. \tag{2.6}$$

Combining (2.5) and (2.6) we have, overall,

$$(A + F)\hat{x} = b, \qquad |F| \leqq f(u)|\hat{L}||\hat{U}|, \qquad f(u) = 4u + 3u^2 + u^3. \tag{2.7}$$

We have avoided using $O(u^2)$ notation in order to emphasize that there are no large constants in the higher-order terms; in particular, $f(u)$ is independent of n.

3. Componentwise backward error and computation of cond (A, x). The backward error result (2.7) applies to arbitrary nonsingular tridiagonal A having an LU factorization. We are interested in determining classes of tridiagonal A for which the bound $|F| \leqq f(u)|\hat{L}||\hat{U}|$ implies the "ideal bound"

$$|F| \leqq g(u)|A|. \tag{3.1}$$

Certainly, (3.1) holds if

$$|\hat{L}||\hat{U}| = |\hat{L}\hat{U}|, \tag{3.2}$$

for then, using (2.5),

$$|\hat{L}||\hat{U}| = |A-E| \leqq |A| + u|\hat{L}||\hat{U}|,$$

so that

$$|\hat{L}||\hat{U}| \leqq \frac{1}{1-u}|A|. \tag{3.3}$$

Three classes of matrices for which (3.2) holds for the exact L and U are identified in the following theorem. A nonsingular $A \in \mathbf{R}^{n\times n}$ is an M-matrix if $a_{ij} \leqq 0$ for all $i \neq j$ and $A^{-1} \geqq 0$. There are many equivalent conditions for A to be an M-matrix [2, Chap. 6]; for example, the condition $A^{-1} \geqq 0$ can be replaced by the condition that all the principal minors of A are positive.

THEOREM 3.1. *Let $A \in \mathbf{R}^{n\times n}$ be nonsingular and tridiagonal. If any of the following conditions hold then A has an LU factorization and $|L||U| = |LU|$:*

(a) *A is symmetric positive definite;*

(b) *A is totally nonnegative, or equivalently, $L \geqq 0$ and $U \geqq 0$;*

(c) *A is an M-matrix, or equivalently, L and U have positive diagonal elements and nonpositive off-diagonal elements;*

(d) *A is sign equivalent to a matrix B of type* (a)–(c); *that is, $A = D_1BD_2$, where $|D_1| = |D_2| = I$.*

Proof. For (a), it is well known that a symmetric positive definite A has an LU factorization in which $U = DL^T$, where D is diagonal with positive diagonal elements. Hence $|L||U| = |L||D||L^T| = |LDL^T| = |LU|$. In (b) and (c) the equivalences, and the existence of an LU factorization, follow from known results on totally nonnegative matrices [4] and M-matrices [2]; $|L||U| = |LU|$ is immediate from the sign properties of L and U. (d) is trivial. □

THEOREM 3.2. *If the tridiagonal matrix A is of type* (a)–(d) *in Theorem* 3.1, *and if the unit roundoff u is sufficiently small, then Gaussian elimination for solving $Ax = b$ succeeds and the computed solution $\hat{x}$ satisfies*

$$(A+F)\hat{x} = b, \qquad |F| \leqq h(u)|A|, \qquad h(u) = \frac{4u+3u^2+u^3}{1-u}. \tag{3.4}$$

Proof. If u is sufficiently small, then for types (a)–(c) the diagonal elements of $\hat{U}$ will be positive, since $\hat{u}_i \to u_i > 0$ as $u \to 0$. It is easy to see that $\hat{u}_i > 0$ for all i ensures that $|\hat{L}||\hat{U}| = |\hat{L}\hat{U}|$. The argument is similar for type (d). The result therefore follows from (2.7) and (3.3). □

Theorem 3.2 appears to be new in the case of M-matrices. A result of the form (3.4) (with a c_n term in the bound) is valid for any totally nonnegative matrix [5]. The symmetric positive definite case in Theorem 3.2 is also known [8].

A corollary of Theorem 3.2 is that it is not necessary to pivot for the matrices specified in the theorem (and, indeed, pivoting could vitiate the bound (3.4)). Note that large multipliers may occur under the conditions of Theorem 3.2, but they do not affect the stability. (Recall the well-known property [21, p. 412] that arbitrarily large multipliers may occur in LU factorization of a general symmetric positive definite matrix, yet the growth factor $\rho_n \leqq 1$.) We stress this point because in [13], which deals with Gaussian elimination of tridiagonal Toeplitz matrices, it is stated that "the stability of the elimination process is controlled by the size of the multipliers m_j." We also mention that the example given by Harrod [14] of the M-matrix

$$A = \begin{bmatrix} 2 & -2 & 0 \\ \varepsilon - 2 & 2 & 0 \\ 0 & -1 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ (\varepsilon-2)/2 & 1 & 0 \\ 0 & -1/\varepsilon & 1 \end{bmatrix} \begin{bmatrix} 2 & -2 & 0 \\ 0 & \varepsilon & 0 \\ 0 & 0 & 3 \end{bmatrix} = LU,$$

for which the multiplier l_{32} is unbounded as $\varepsilon \to 0$, is an example where Gaussian elimination performs very stably, as Theorem 3.2 shows.

We now turn our attention to computing cond (A, x). We show that if $|L||U| = |LU|$ then cond (A, x) can be computed in $O(n)$ operations.

THEOREM 3.3. *If the nonsingular tridiagonal matrix $A \in \mathbf{R}^{n\times n}$ has the LU factorization $A = LU$ and $|L||U| = |A|$, then $|U^{-1}||L^{-1}| = |A^{-1}|$.*

Proof. Using the notation of (2.1) and (2.2), $|L||U| = |A| = |LU|$ if and only if for all i

$$|l_i e_{i-1} + u_i| = |l_i||e_{i-1}| + |u_i|,$$

that is, if

$$\operatorname{sign}\left(\frac{l_i e_{i-1}}{u_i}\right) = 1. \tag{3.5}$$

Using the formulae

$$(U^{-1})_{ij} = \frac{1}{u_j}\prod_{p=i}^{j-1}\left(\frac{-e_p}{u_p}\right) \qquad (j \geqq i), \tag{3.6}$$

$$(L^{-1})_{ij} = \prod_{p=j}^{i-1}(-l_{p+1}) \qquad (i \geqq j), \tag{3.7}$$

we have

$$\begin{aligned}
(U^{-1}L^{-1})_{ij} &= \sum_{k=\max(i,j)}^{n} (U^{-1})_{ik}(L^{-1})_{kj} \\
&= \sum_{k=\max(i,j)}^{n} \frac{1}{u_k}\prod_{p=i}^{k-1}\left(\frac{-e_p}{u_p}\right)\prod_{p=j}^{k-1}(-l_{p+1}) \\
&= \prod_{p=i}^{\max(i,j)-1}\left(\frac{-e_p}{u_p}\right)\cdot\prod_{p=j}^{\max(i,j)-1}(-l_{p+1})\cdot\sum_{k=\max(i,j)}^{n}\frac{1}{u_k}\prod_{p=\max(i,j)}^{k-1}\left(\frac{e_p l_{p+1}}{u_p}\right) \\
&= \prod_{p=i}^{\max(i,j)-1}\left(\frac{-e_p}{u_p}\right)\cdot\prod_{p=j}^{\max(i,j)-1}(-l_{p+1})\cdot\frac{1}{u_{\max(i,j)}}\cdot\sum_{k=\max(i,j)}^{n}\prod_{p=\max(i,j)}^{k-1}\left(\frac{e_p l_{p+1}}{u_{p+1}}\right).
\end{aligned}$$

Thus, in view of (3.5), it is clear that $|U^{-1}L^{-1}|_{ij} = (|U^{-1}||L^{-1}|)_{ij}$, as required. □

To see the significance of the property $|U^{-1}||L^{-1}| = |A^{-1}|$, note first that, as is clear from (3.6) and (3.7),

$$|U^{-1}| = M(U)^{-1}, \qquad |L^{-1}| = M(L)^{-1},$$

where for $B \in \mathbf{R}^{n\times n}$ the comparison matrix $M(B)$ is defined by

$$(M(B))_{ij} = \begin{cases} |b_{ii}|, & i = j, \\ -|b_{ij}|, & i \neq j. \end{cases}$$

Thus, if $|A^{-1}| = |U^{-1}||L^{-1}|$ and $y \geqq 0$ then

$$|A^{-1}|y = |U^{-1}||L^{-1}|y = M(U)^{-1}M(L)^{-1}y.$$

By taking $y = |A||x|$ it follows that cond (A, x) can be computed in $O(n)$ operations:

$$\begin{aligned} &\text{form } y = |A||x|, \\ &\text{solve } M(L)v = y, \\ &\text{solve } M(U)w = v, \\ &\text{compute } \|w\|_\infty / \|x\|_\infty. \end{aligned} \tag{3.8}$$

For the special case $y = e$ and A symmetric positive definite, (3.8) was used in [15, § 6] to compute $\|A^{-1}\|_\infty$ in $O(n)$ operations.

Of course, in practice we use the computed $\hat{L}$ and $\hat{U}$ in place of the exact LU factors. If cond (A) is not too large (cond $(A)u < \frac{1}{2}$, say), then we are guaranteed a satisfactory computed value of cond (A, x), that is, one having some correct digits.

4. Diagonally dominant matrices. A in (2.1) is diagonally dominant by rows if

$$|d_i| \geqq |c_i| + |e_i| \quad \text{for all } i \quad (c_1 \equiv e_n \equiv 0),$$

and diagonally dominant by columns if A^T is diagonally dominant by rows. Such A have an LU factorization, but $|L||U| \neq |A|$ in general, and so we cannot apply the results of the last section. However, as the next result shows, $|L||U|$ can be bounded by a small multiple of $|A|$. Combining this result with (2.7), we are able to conclude that the componentwise backward error is small in solving a diagonally dominant tridiagonal system $Ax = b$.

THEOREM 4.1. *Suppose $A \in \mathbf{R}^{n\times n}$ is nonsingular, tridiagonal, and diagonally dominant by rows or columns, and let A have the LU factorization $A = LU$. Then $|L||U| \leqq 3|A|$.*

Proof. If $|i - j| = 1$ then $(|L||U|)_{ij} = |a_{ij}|$, so it suffices to consider the diagonal elements and show that (using the notation of (2.2))

$$|l_i e_{i-1}| + |u_i| \leqq 3|d_i|.$$

The rest of the proof is for the case where A is diagonally dominant by rows; the proof for diagonal dominance by columns is similar.

First, we claim that $|e_i| \leqq |u_i|$ for all i. The proof is by induction. For $i = 1$ the result is immediate, and if it is true for $i - 1$ then from (2.3)

$$\begin{aligned} |u_i| &\geqq |d_i| - |l_i||e_{i-1}| = |d_i| - \frac{|c_i|}{|u_{i-1}|}|e_{i-1}| \\ &\geqq |d_i| - |c_i| \geqq |e_i|, \end{aligned}$$

as required. Note that, similarly, $|u_i| \leqq |d_i| + |c_i|$. Finally,

$$|l_i e_{i-1}| + |u_i| = \left|\frac{c_i}{u_{i-1}} e_{i-1}\right| + |u_i| \leqq |c_i| + |u_i|$$
$$\leqq |c_i| + (|d_i| + |c_i|)$$
$$\leqq 3|d_i|. \qquad \square$$

Unfortunately, it is not generally true for diagonally dominant A that $|A^{-1}| = |U^{-1}||L^{-1}|$, so we cannot compute cond (A, x) using the $O(n)$ operations technique of the last section. However we can compute the upper bound in

$$|A^{-1}|y \leqq |U^{-1}||L^{-1}|y \qquad (y = |A||x|)$$

in $O(n)$ operations. Concentrating, for the moment, on diagonal dominance by rows, a bound for how much of an overestimate this upper bound can be is provided by the following result.

THEOREM 4.2. *Suppose the nonsingular, row diagonally dominant, tridiagonal matrix $A \in \mathbf{R}^{n\times n}$ has the LU factorization $A = LU$. Then, if $y \geqq 0$,*

$$\||U^{-1}||L^{-1}|y\|_\infty \leqq (2n-1)\||A^{-1}|y\|_\infty.$$

Proof. We have $L^{-1} = UA^{-1}$, so

$$|U^{-1}||L^{-1}|y \leqq |U^{-1}||U||A^{-1}|y = |V^{-1}||V||A^{-1}|y,$$

where the bidiagonal matrix $V = \operatorname{diag}(u_{ii})^{-1}U$ has $v_{ii} \equiv 1$ and $|v_{i,i+1}| = |e_i/u_i| \leqq 1$ (see the proof of Theorem 4.1). Thus

$$|U^{-1}||L^{-1}|y \leqq \begin{bmatrix} 1 & 1 & \cdots & 1 \\ & 1 & \cdots & 1 \\ & & \ddots & \vdots \\ & & & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & & \\ & 1 & \ddots & \\ & & \ddots & 1 \\ & & & 1 \end{bmatrix} |A^{-1}|y,$$

and the result follows on taking norms. $\square$

Theorem 4.2 says that when A is row diagonally dominant our upper bound for cond (A, x) is too big by a factor at most $2n - 1$. This is somewhat unsatisfactory since n can be very large. For $n = 2$ the bound in Theorem 4.2 is attained as $\alpha \to \infty$ in the example

$$A = \begin{bmatrix} 1 & 1 \\ -\alpha & \alpha+1 \end{bmatrix}, \qquad y = e.$$

For general n we have not been able to construct any examples in which the bound in Theorem 4.2 is attained (except by relaxing the row diagonal dominance assumption). In a wide variety of numerical tests with both random and nonrandom matrices, the upper bound has never exceeded the quantity it bounds by more than a small constant factor (3, say). Moreover, the bound is exact if the row diagonally dominant A happens to be symmetric (so that it is positive definite), nonnegative (that is, $A \geqq 0$, which implies it is totally nonnegative), or an M-matrix—all three cases are common in applications. We therefore regard the upper bound as reliable in practice, and conjecture that the factor $2n - 1$ in Theorem 4.2 can be improved to a constant independent of n.

We mention that Neumaier [18] found that $\||U^{-1}||L^{-1}|y\|_\infty \leqq 2\||A^{-1}|y\|_\infty$ held in a small number of tests with *full* random row diagonally dominant matrices and random $y > 0$, and this inequality is confirmed by our own tests with random matrices.

However, no theoretical bound on the overestimation factor is known in the case of full A.

A weaker analogue of Theorem 4.2 holds when A is diagonally dominant by columns. The inequality $|U^{-1}||L^{-1}|y \leqq |A^{-1}||L||L^{-1}|y$ leads to, for $y > 0$,

$$\||U^{-1}||L^{-1}|y\|_\infty \leqq (2n-1)\theta\||A^{-1}|y\|_\infty, \qquad \theta = \frac{\max_i |y_i|}{\min_i |y_i|}.$$

Despite the unbounded θ term in this inequality, we have not observed or constructed any examples where the upper bound is more than a small constant factor too big. Thus we regard the upper bound as being of practical use also when A is diagonally dominant by columns.

5. General tridiagonal matrices. We turn now to tridiagonal systems $Ax = b$ where A does not fall into any of the classes considered in the previous two sections. Suppose GEPP is used to solve the system. Suppose also that we wish to refer to backward and forward error bounds of the forms (1.1) and (1.2) and to estimate or compute $\kappa_\infty(A)$. Several algorithms for computing $\kappa_\infty(A)$ exactly in $O(n)$ operations are presented in [15]. As explained in [15], these algorithms (except the algorithm for symmetric positive definite A) have the property that the intermediate numbers can have a large dynamic range (the more so, the more diagonally dominant A is), and the algorithms can break down in floating-point arithmetic due to underflow or overflow. These numerical problems can be overcome, but at a nontrivial increase in cost (see [15]). Our preferred approach is to use the matrix norm estimator SONEST from [16]. This provides an *estimate* for $\|B\|_1$ (a lower bound) at the cost of computing a few matrix-vector products Bc and B^Td. Typically four or five products are required; the norm estimate is frequently exact and is almost always correct to within a factor 3. In our application, $B = A^{-T}$, and so we need to solve a few linear systems $A^Ty = c$ and $Az = d$, which can be done using the LU factorization already computed. The SONEST approach has about the same computational cost as the methods in [15].

Next, suppose that GEPP followed by iterative refinement is used to solve the tridiagonal system $Ax = b$. Then, under suitable assumptions, a result of the form (3.4) holds [1], [20], and so the appropriate condition number is cond (A, x). (See [1] for a discussion of possible violation of the assumptions when x and b are sparse, and for suggested cures.) The techniques of [15] could be adapted to compute cond (A, x) in $O(n)$ operations, with the same practical numerical difficulties described above. However, as shown in [1], [7], SONEST can be used to estimate cond (A, x) (even for general A), and this is the approach we recommend.

Finally, note that for GEPP one could use the elementwise backward error result (2.7) (suitably modified to take account of pivoting), for which a forward error bound involving the condition number $\||A^{-1}||\hat{L}||\hat{U}||x|\|_\infty/\|x\|_\infty$ can be derived. Again, this condition number (which is row scaling independent) can be estimated using SONEST.

6. Practical considerations. We discuss several practical issues concerning the condition numbers and algorithms described above.

- For symmetric positive definite A the standard way to solve $Ax = b$ is by using a Cholesky or LDL^T factorization, rather than an LU factorization. The LINPACK routine SPTSL uses a nonstandard "LUB" factorization resulting from the BABE ("burn at both ends") algorithm (see [10], [15]). The results of § 3 are applicable to all of these factorizations, with minor modifications. Note that the LDL^T factorization requires n fewer divisions in the substitution stage than the Cholesky factorization.
- A drawback to the computation or estimation of cond $(A, x) = \||A^{-1}||A||x|\|_\infty/\|x\|_\infty$ is the need to keep a copy of A in order to form the product $|A||x|$ once x has

been computed. If n is large it may not be possible to store a copy of A. One can circumvent this problem for the matrices in Theorems 3.1 and 4.1, for which $|A| = |L||U|$ and $|A| \leq |L||U| \leq 3|A|$, respectively, by using $|L||U||x|$ in place of $|A||x|$.

- We give the computational costs of the error estimation techniques in two particular cases, in terms of flops [12, p. 32]. For a general tridiagonal $A \in \mathbf{R}^{n\times n}$, factoring $PA = LU$ by GEPP and solving $Ax = b$ by substitution costs $(5 + 2p)n$ flops, where $p \in [0, 1]$ depends on the number of interchanges; estimating $\kappa_\infty(A)$ requires $2n$ flops to compute $\|A\|_\infty$ and, typically, $4(3 + p)n$ or $5(3 + p)n$ flops to estimate $\|A^{-1}\|_\infty$ using SONEST. For a symmetric positive definite $A \in \mathbf{R}^{n\times n}$, factoring $A = LDL^T$ and solving $Ax = b$ requires $5n$ flops, and computing cond (A, x) requires $6n$ flops. Thus these error estimation techniques at least double the cost of solving a linear system.
- Instead of computing cond (A, x) one could compute cond (A) = cond $(A, e) \geq$ cond (A, x). The same cond (A) value could be reused when solving systems with the same A but different right-hand sides. However, this approach reduces the sharpness of the bounds, since cond (A)/cond (A, x) can be arbitrarily large.

Finally, we present a numerical experiment that gives an indication of the sharpness of the various error bounds. We used a tridiagonal matrix given by Dorr [11] that occurs in the solution of a singular perturbation problem by finite differences. With $m = \lfloor (n + 1)/2 \rfloor$, $h = 1/(n + 1)$, and $\varepsilon > 0$, the matrix is defined by (see (2.1))

$$c_i = \begin{cases} -\varepsilon/h^2, & 1 \leq i \leq m, \\ -\varepsilon/h^2 + (\tfrac{1}{2} - ih)/h^2, & m+1 \leq i \leq n, \end{cases}$$

$$e_i = \begin{cases} -\varepsilon/h^2 - (\tfrac{1}{2} - ih)/h^2, & 1 \leq i \leq m, \\ -\varepsilon/h^2, & m+1 \leq i \leq n, \end{cases}$$

and $d_i = -(c_i + e_i)$, $1 \leq i \leq n$ (note that c_1 and e_n are introduced solely to define d_1 and d_n). A is a nonsingular, row diagonally dominant M-matrix. For small values of the parameter ε the matrix is ill-conditioned.

We chose $n = 50$ and $\varepsilon = 0.009$. We solved $Ax = b$ for six different right-hand sides. The computations were done in PC-MATLAB, with simulated single precision arithmetic of unit roundoff $u = 2^{-23} \approx 1.2 \times 10^{-7}$. For each system we computed $\hat{x}$ in single precision and x and the relative error $\|x - \hat{x}\|_\infty / \|x\|_\infty$ in double precision. Since A is an M-matrix, cond (A, x), cond (A), and $\kappa_\infty(A)$ were computed in $O(n)$ flops according to (3.8) (using $y = e$ to compute $\kappa_\infty(A)$). The results are given in Table 6.1.

For our test problem, (1.5) takes the form (using (3.4))

$$\frac{\|x - \hat{x}\|_\infty}{\|x\|_\infty} \leq 10.9 \text{ cond } (A, x)u. \tag{6.1}$$

TABLE 6.1
Numerical results, n = 50.

cond(A) = 1.33E6, $\kappa_\infty(A)$ = 1.85E6
$p = e_n + e_{n-1} + \cdots + e_{n-4}$
$q = (1, \alpha, \alpha^2, \cdots, 10^{-5})$, $\alpha = 10^{-5/(n-1)}$
rand = vector with random elements from uniform $[-1, 1]$ distribution

	$x = p$	$x = e_1$	$x = q$	$x = e$	x = rand	$b = e_n$
cond(A, x)	1.73E2	3.82E0	8.89E3	1.33E6	5.50E5	8.87E5
$\dfrac{\|x - \hat{x}\|_\infty}{u\|x\|_\infty}$	1.25E0	0.00E0	9.92E2	1.42E2	1.75E4	1.09E2

In the traditional bound (1.2) there is a similar constant and cond (A, x) is replaced by $\kappa_\infty(A)$. From Table 6.1 we see that in the first three cases cond (A, x) is significantly smaller than cond (A) and $\kappa_\infty(A)$; this indicates the value of using a condition number that depends on x. The bound (6.1) is of variable sharpness, but it is always smaller than the traditional bound.

Acknowledgment. Des Higham helped to polish the presentation.

REFERENCES

[1] M. ARIOLI, J. W. DEMMEL, AND I. S. DUFF, *Solving sparse linear systems with sparse backward error*, SIAM J. Matrix Anal. Appl., 10 (1989), pp. 165–190.

[2] A. BERMAN AND R. J. PLEMMONS, *Nonnegative Matrices in the Mathematical Sciences*, Academic Press, New York, 1979.

[3] C. H. BISCHOF, J. W. DEMMEL, J. J. DONGARRA, J. J. DU CROZ, A. GREENBAUM, S. J. HAMMARLING, AND D. C. SORENSEN, Provisional contents, LAPACK Working Note No. 5, Report ANL-88-38, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, 1988.

[4] C. W. CRYER, *The LU-factorization of totally positive matrices*, Linear Algebra Appl., 7 (1973), pp. 83–92.

[5] C. DE BOOR AND A. PINKUS, *Backward error analysis for totally positive linear systems*, Numer. Math., 27 (1977), pp. 485–490.

[6] J. W. DEMMEL, J. J. DONGARRA, J. J. DU CROZ, A. GREENBAUM, S. J. HAMMARLING, AND D. C. SORENSEN, *Prospectus for the development of a linear algebra library for high-performance computers*, Tech. Memorandum No. 97, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, 1987.

[7] J. W. DEMMEL, J. J. DU CROZ, S. J. HAMMARLING, AND D. C. SORENSEN, *Guidelines for the design of symmetric eigenroutines, SVD, and iterative refinement and condition estimation for linear systems*, LAPACK Working Note No. 4, Tech. Memorandum 111, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, 1988.

[8] J. W. DEMMEL AND W. KAHAN, *Computing small singular values of bidiagonal matrices with guaranteed high relative accuracy*, LAPACK Working Note No. 3, Tech. Memorandum 110, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, 1988; SIAM J. Sci. Statist. Comput., 11 (1990), to appear.

[9] J. J. DONGARRA, Private communication, 1988.

[10] J. J. DONGARRA, J. R. BUNCH, C. B. MOLER, AND G. W. STEWART, LINPACK *Users' Guide*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1979.

[11] F. W. DORR, *An example of ill-conditioning in the numerical solution of singular perturbation problems*, Math. Comp., 25 (1971), pp. 271–283.

[12] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, The Johns Hopkins University Press, Baltimore, MD, 1983.

[13] M. D. GUNZBURGER AND R. A. NICOLAIDES, *Stability of Gaussian elimination without pivoting on tridiagonal Toeplitz matrices*, Linear Algebra Appl., 45 (1982), pp. 21–28.

[14] W. J. HARROD, *LU-decompositions of tridiagonal irreducible H-matrices*, SIAM J. Algebraic Discrete Methods, 7 (1986), pp. 180–187.

[15] N. J. HIGHAM, *Efficient algorithms for computing the condition number of a tridiagonal matrix*, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 150–165.

[16] ———, *Algorithm* 674: FORTRAN *codes for estimating the one-norm of a real or complex matrix, with applications to condition estimation*, ACM Trans. Math. Software, 14 (1988), pp. 381–396.

[17] ———, *The accuracy of solutions to triangular systems*, SIAM J. Numer. Anal., 26 (1989), pp. 1252–1265.

[18] A. NEUMAIER, *On the comparison of H-matrices with M-matrices*, Linear Algebra Appl., 83 (1986), pp. 135–141.

[19] R. D. SKEEL, *Scaling for numerical stability in Gaussian elimination*, J. Assoc. Comput. Mach., 26 (1979), pp. 494–526.

[20] ———, *Iterative refinement implies numerical stability for Gaussian elimination*, Math. Comp., 35 (1980), pp. 817–832.

[21] G. W. STEWART, *Introduction to Matrix Computations*, Academic Press, New York, 1973.

[22] J. H. WILKINSON, *Rounding Errors in Algebraic Processes*, Notes on Applied Science No. 32, Her Majesty's Stationery Office, London, 1963.