ESTIMATES OF OPTIMAL BACKWARD PERTURBATIONS FOR LINEAR LEAST SQUARES PROBLEMS*

JOSEPH F. GRCAR†, MICHAEL A. SAUNDERS‡, AND ZHENG SU§

Abstract. Numerical tests are used to validate a practical estimate for the optimal backward errors of linear least squares problems. This solves a thirty-year-old problem suggested by Stewart and Wilkinson.

Key words. linear least squares, backward errors, optimal backward error

AMS subject classifications. 65F20, 65F50, 65G99

> "A great deal of thought, both by myself and by J. H. Wilkinson, has not solved this problem, and I therefore pass it on to you: *find easily computable statistics that are both necessary and sufficient for the stability of a least squares solution.*" — G. W. Stewart [24, pp. 6–7]

1. Introduction. Our aim is to examine the usefulness of a certain quantity as a practical backward error estimator for the least squares (LS) problem:

$$\min_x \|Ax - b\|_2 \qquad \text{where} \qquad b \in \mathbb{R}^m \text{ and } A \in \mathbb{R}^{m\times n}.$$

Throughout the paper, x denotes an *arbitrary vector* in $\mathbb{R}^n$. If any such x solves the LS problem for data $A + \delta A$, then the perturbation δA is called a *backward error* for x. This name is borrowed from the context of Stewart and Wilkinson's remarks, backward rounding error analysis, which finds and bounds some δA when x is a computed solution. When x is arbitrary, it may be more appropriate to call δA a "data perturbation" or a "backward perturbation" rather than a "backward *error*." All three names have been used in the literature.

The size of the smallest backward error is $\mu(x) = \min_{\delta A} \|\delta A\|_{\mathrm{F}}$. A precise definition and more descriptive notation for this are

$$\mu(x) \;=\; \left\{ \begin{array}{l} \text{the size of data perturbation, for matrices in least squares} \\ \text{problems, that is optimally small in the Frobenius norm,} \\ \text{as a function of the approximate solution } x \end{array} \right\} \;=\; \mu_{\mathrm{F}}^{(\mathrm{LS})}(x)\,.$$

This level of detail is needed here only twice, so we usually abbreviate it to "optimal backward error" and write $\mu(x)$. The concept of optimal backward error originated with Oettli and Prager [19] in the context of linear equations.

If $\mu(x)$ can be estimated or evaluated inexpensively, then the literature describes three uses for it.

1. *Accuracy criterion.* When the data of a problem have been given with an error that is greater than $\mu(x)$, then x must be regarded as solving the problem, to the extent the problem is known. Conversely, if $\mu(x)$ is greater than the

*Version 11 draft, September 18, 2007. Submitted to SIMAX.

†Center for Computational Sciences and Engineering, Lawrence Berkeley National Lab, Berkeley, CA 94720-8142 (jfgrcar@lbl.gov). The work of this author was supported by the U.S. Department of Energy under contract no. DE-AC02-05-CH11231.

‡Department of Management Science and Engineering, Stanford University, Stanford, CA 94305-4026 (saunders@stanford.edu). The research of this author was supported by National Science Foundation grant CCR-0306662 and Office of Naval Research grant N00014-02-1-0076.

§Department of Applied Mathematics and Statistics, SUNY Stony Brook, Stony Brook, NY 11733 (zhengsu@ams.sunysb.edu).

uncertainty in the data, then x must be rejected. These ideas originated with John von Neumann and Herman Goldstine [18] and were rediscovered by Oettli and Prager.

2. *Run-time stability estimation.* A calculation that produces x with small $\mu(x)$ is called backwardly stable. Stewart and Wilkinson [24, pp. 6–7], Karlson and Waldén [15, p. 862] and Malyshev and Sadkane [16, p. 740] emphasized the need for "practical" and "accurate and fast" ways to determine $\mu(x)$ for least squares problems.
3. *Exploring the stability of new algorithms.* Many fast algorithms have been developed for LS problems with various kinds of structure. Gu [12, p. 365] [13] explained that it is useful to examine the stability of such algorithms without having to perform backward error analyses of them.

When x is a computed solution, Wilkinson would have described these uses for $\mu(x)$ as "a posteriori" rounding error analyses.

The exact value of $\mu(x)$ was discovered by Waldén, Karlson and Sun [28] in 1995. To evaluate it, they recommended a formula that Higham had derived from their pre-publication manuscript [28, p. 275] [14, p. 393],

$$\mu(x) \;=\; \min\left\{ \frac{\|r\|}{\|x\|}\,,\, \sigma_{\min}[A\ \ B] \right\}, \qquad B \;=\; \frac{\|r\|}{\|x\|}\left(I - \frac{rr^t}{\|r\|^2} \right), \tag{1.1}$$

where $r = b - Ax$ is the residual for the approximate solution, $\sigma_{\min}$ is the smallest singular value of the $m \times (n+m)$ matrix in brackets, and $\|\cdot\|$ means the 2-norm unless otherwise specified. There are similar formulas when both A and b are perturbable, but they are not discussed here. It is interesting to note that a prominent part of these formulas is the optimal backward error of the linear equations $Ax = b$, namely

$$\eta(x) \equiv \frac{\|r\|}{\|x\|} \;=\; \mu_{\mathrm{F}}^{(\mathrm{LE})}(x) \;=\; \mu_2^{(\mathrm{LE})}(x)\,. \tag{1.2}$$

The singular value in (1.1) is expensive to calculate by dense matrix methods, so other ways to obtain the backward error have been sought. Malyshev and Sadkane [16] proposed an iterative process based on the Golub-Kahan process [8] (often called Lanczos bidiagonalization).

Other authors including Waldén, Karlson and Sun have derived explicit *approximations* for $\mu(x)$. One estimate in particular has been studied in various forms by Karlson and Waldén [15], Gu [12], and Grcar [11]. It can be written as

$$\widetilde{\mu}(x) \;=\; \left\| \left(\|x\|^2 A^t A + \|r\|^2 I \right)^{-1/2} A^t r \right\| \;=\; \left\| \left(A^t A + \eta^2 I \right)^{-1/2} A^t r \right\| / \|x\|\,. \tag{1.3}$$

For this quantity:

- Karlson and Waldén showed [15, p. 864, eqn. 2.5 with $y = y_{\mathrm{opt}}$] that, in the notation of this paper,

$$\frac{2}{2+\sqrt{2}}\,\widetilde{\mu}(x) \;\le\; f(y_{\mathrm{opt}})\,,$$

  where $f(y_{opt})$ is a complicated expression that is a lower bound for the smallest backward error in the spectral norm, $\mu_2^{(\mathrm{LS})}(x)$. It is also a lower bound for $\mu(x) = \mu_{\mathrm{F}}^{(\mathrm{LS})}(x)$ because $\|\delta A\|_2 \le \|\delta A\|_{\mathrm{F}}$. Therefore Karlson and Waldén's inequality can be rearranged to

$$\frac{\widetilde{\mu}(x)}{\mu(x)} \;\le\; \frac{2+\sqrt{2}}{2} \;\approx\; 1.707\,. \tag{1.4}$$

- Gu [12, p. 367, cor. 2.2] established the bounds

$$\frac{\|r_*\|}{\|r\|} \;\le\; \frac{\widetilde{\mu}(x)}{\mu(x)} \;\le\; \frac{\sqrt{5}+1}{2} \;\approx\; 1.618\,, \tag{1.5}$$

  where r_* is the unique, true residual of the LS problem. He used these inequalities to prove a theorem about the definition of numerical stability for LS problems. Gu derived the bounds assuming that A has full column rank. The lower bound in (1.5) should be slightly less than 1 because it is always true that $\|r_*\| \le \|r\|$, and because $r \approx r_*$ when x is a good approximation to a solution.
- Finally, Grcar [11, thm. 4.4], based on [10], proved that $\widetilde{\mu}(x)$ asymptotically equals $\mu(x)$ in the sense that

$$\lim_{x \to x_*} \frac{\widetilde{\mu}(x)}{\mu(x)} \;=\; 1\,, \tag{1.6}$$

  where x_* is any solution of the LS problem. The hypotheses for this are that A, r_*, and x_* are not zero. This limit and both equations (1.1) and (1.3) do not restrict the rank of A or the relative sizes of m and n.

All these bounds and limits suggest that (1.3) is a robust estimate for the optimal backward error of least squares problems. However, this formula has not been examined numerically. It receives only brief mention in the papers of Karlson and Waldén, and Gu, and neither they nor Grcar performed numerical experiments with it. The aim of this paper is to determine whether $\widetilde{\mu}(x)$ is an acceptable estimate for $\mu(x)$ in practice, thereby answering Stewart and Wilkinson's question.

2. When both A and b are perturbed. A practical estimate for the optimal backward error when both A and b are perturbed is also of interest. In this case, the optimal backward error is defined as

$$\min_{\Delta A, \Delta b} \{\|\Delta A, \theta \Delta b\|_F : \|(A+\Delta A)y - (b+\Delta b)\|_2 = \min\},$$

where θ is a weighting parameter. (Taking the limit $\theta \to \infty$ forces $\Delta b = 0$, giving the case where only A is perturbed.) From Waldén et al. [28] and Higham [14, p. 393] the exact backward error is

$$\mu_{A,b}(x) = \min\{\sqrt{\nu}\eta,\ \sigma_{\min}[A \;\; B]\},$$

where

$$\eta = \frac{\|r\|}{\|x\|}, \qquad B = \sqrt{\nu}\eta\left(I - \frac{rr^t}{\|r\|^2}\right), \qquad \nu = \frac{\theta^2\|x\|^2}{1+\theta^2\|x\|^2}.$$

Thus, $\mu_{A,b}(x)$ is the same as $\mu(x)$ in (1.1) with η changed to $\bar{\eta} \equiv \sqrt{\nu}\eta$ (as noted by Su [25]). Hence we can estimate $\mu_{A,b}(x)$ using methods for estimating $\mu(x)$, replacing $\eta = \|r\|/\|x\|$ by $\bar{\eta} = \sqrt{\nu}\|r\|/\|x\|$. In particular, from (1.3) we see that $\mu_{A,b}(x)$ may be estimated by

$$\widetilde{\mu}_{A,b}(x) \;=\; \left\|\left(A^tA + \bar{\eta}^2 I\right)^{-1/2} A^t r\right\| / \|x\|\,. \tag{2.1}$$

The asymptotic property (1.6) also follows because $\widetilde{\mu}(x)$ and $\widetilde{\mu}_{A,b}(x)$ have the same essential structure.

In the remainder of the paper, we consider methods for computing $\widetilde{\mu}(x)$ when only A is perturbed, but the same methods apply to computing $\widetilde{\mu}_{A,b}(x)$.

3. The KW problem and projections. Karlson and Waldén [15, p. 864] draw attention to the full-rank LS problem

$$K = \begin{bmatrix} A \\ \frac{\|r\|}{\|x\|} I \end{bmatrix}, \qquad v = \begin{bmatrix} r \\ 0 \end{bmatrix}, \qquad \min_y \|Ky - v\|, \tag{3.1}$$

which proves central to the computation of $\widetilde{\mu}(x)$. It should be mentioned that LS problems with this structure are called "damped", and have been studied in the context of Tikhonov regularization of ill-posed LS problems [3, pp. 101–102]. We need to study *three* such systems involving various A and r. To do so, we need some standard results on QR factorization and projections. We state these in terms of a full-rank LS problem $\min_y \|Ky - v\|$ with general K and v.

LEMMA 3.1. *Suppose the matrix K has full column rank and QR factorization*

$$K = Q \begin{bmatrix} R \\ 0 \end{bmatrix} = YR, \qquad Q = \begin{bmatrix} Y & Z \end{bmatrix}, \tag{3.2}$$

where R is upper triangular and nonsingular, and Q is square and orthogonal, so that $Y^tY = I$, $Z^tZ = I$, and $YY^t + ZZ^t = I$. The associated projection operators may be written as

$$P = K(K^tK)^{-1}K^t = YY^t, \qquad I - P = ZZ^t. \tag{3.3}$$

LEMMA 3.2. *For the quantities in Lemma 3.1, the LS problem $\min_y \|Ky - v\|$ has a unique solution and residual vector defined by $Ry = Y^tb$ and $t = v - Ky$, and the two projections of v satisfy*

$$Pv = Ky = YY^tv, \qquad \|Ky\| = \|Y^tv\|, \tag{3.4}$$

$$(I - P)v = \quad t = ZZ^tv, \qquad \|t\| = \|Z^tv\|. \tag{3.5}$$

(We do not need (3.5), but it is included for completeness.)

We now find that $\widetilde{\mu}(x)$ in (1.3) is the norm of a certain vector's projection. Let K and v be as in the KW problem (3.1). From (1.3) and the definition of P in (3.3) we see that $\|x\|^2\,\widetilde{\mu}(x)^2 = v^tPv$, and from (3.4) we have $v^tPv = \|Y^tv\|^2$. It follows again from (3.4) that

$$\widetilde{\mu}(x) = \frac{\|Pv\|}{\|x\|} = \frac{\|Y^tv\|}{\|x\|} = \frac{\|Ky\|}{\|x\|}, \tag{3.6}$$

where Y and $\|Y^tv\|$ may be obtained from the *reduced* QR factorization $K = YR$ in (3.2). (It is not essential to keep the Z part of Q.) Alternatively, $\|Ky\|$ may be obtained after the KW problem is solved by *any* method.

4. Evaluating the estimate. Several ways to solve $\min_x \|Ax - b\|_2$ produce matrix factorizations that can be used to evaluate $\widetilde{\mu}(x)$ in (1.3) or (3.6) efficiently. We describe some of these methods here. If x is arbitrary, then the same procedures may still be used to evaluate $\widetilde{\mu}(x)$ at the extra cost of calculating the factorizations just for this purpose.

TABLE 4.1
Operation counts of solving LS problems by SVD methods with and without forming $\widetilde{\mu}(x)$. The work to evaluate $\widetilde{\mu}(x)$ includes that of r. Only leading terms are shown.

task	operations	source
form U, Σ, V by Chan SVD	$6mn^2 + 20n^3$	[9, p. 239]
solve LS given U, Σ, V	$2mn + 2n^2$	
evaluate $\widetilde{\mu}(x)$ by (4.1) given U, Σ, V	$4mn + 10n$	
solve LS by Chan SVD	$2mn^2 + 11n^3$	[9, p. 248]

4.1. SVD methods. A formula essentially due to Gu [12] can evaluate $\widetilde{\mu}(x)$ when a singular value decomposition (SVD) is used to solve the LS problem. For purposes of this derivation, such a decomposition without restrictions on m and n is $A = U\Sigma V^t$ where Σ is a square matrix and where U and V have orthonormal columns but may not be square. With this notation it follows that

$$\begin{aligned}
\|x\|\,\widetilde{\mu}(x) &= \left\|\left(A^tA + \eta^2 I\right)^{-1/2} A^t r\right\| \\
&= \left\|\left(V\Sigma^2 V^t + \eta^2 I\right)^{-1/2} V\Sigma\, U^t r\right\| \\
&= \left\|\left[V\left(\Sigma^2 + \eta^2 I\right)V^t\right]^{-1/2} V\Sigma\, U^t r\right\| \\
&= \left\|V\left(\Sigma^2 + \eta^2 I\right)^{-1/2} V^t V\Sigma\, U^t r\right\| \\
&= \left\|\left(\Sigma^2 + \eta^2 I\right)^{-1/2} \Sigma\, U^t r\right\| , \qquad (4.1)
\end{aligned}$$

where $\eta = \eta(x)$ in (1.2). Note that the dimension of I may change from line 2 to 3: since the matrix in the square root is applied only to the columns of V, it is possible to pull V and V^t outside the sum even when $I \neq VV^t$. Equation (4.1) is roughly how Gu [12, p. 367, cor. 2.2] stated the estimate in the full rank case.

Calculating $\widetilde{\mu}(x)$ has negligible cost once U, Σ and V have been formed. However, the most efficient SVD algorithms for LS problems accumulate $U^t b$ rather than form U. This saving cannot be realized when U is needed to evaluate $\widetilde{\mu}(x)$. As a result, Table 4.1 shows the operations triple from roughly $2mn^2$ for x, to $6mn^2$ for both x and $\widetilde{\mu}(x)$. This is still much less than the cost of evaluating the exact $\mu(x)$ by (1.1) because about $4m^3 + 2m^2 n$ arithmetic operations are needed to find all singular values of an $m \times (n+m)$ matrix [9, p. 239].

4.2. QR methods. If QR factors of A are available (e.g., from solving the original LS problem), the required projection may be evaluated in two stages. Let the factors be denoted by subscript A. Applying Y_A^t to the top parts of K and v yields an equivalent LS problem

$$K' = \begin{bmatrix} R_A \\ \frac{\|r\|}{\|x\|} I \end{bmatrix}, \qquad v' = \begin{bmatrix} Y_A^t r \\ 0 \end{bmatrix}, \qquad \min_y \|K'y - v'\| . \qquad (4.2)$$

If A has low column rank, we would still regard R_A and Y_A as having n columns. A second QR factorization gives

$$\widetilde{\mu}(x) \;=\; \frac{\|Y_{K'}^t\, v'\|}{\|x\|} . \qquad (4.3)$$

TABLE 4.2

Operation counts of solving LS problems by QR methods and then evaluating $\widetilde{\mu}(x)$ when $m \geq n$. The work to evaluate $Y_A^t r$ includes that of r. Only leading terms are shown.

task	operations	source
solve LS by Householder QR, retaining Y_A	$2mn^2$	[9, p. 248]
form $Y_A^t r$ and v'	$4mn$	
apply $Y_{K'}^t$ to v'	$\frac{8}{3}n^3$	[15, p. 864]
finish evaluating $\widetilde{\mu}(x)$ by (4.3)	$2n$	

TABLE 4.3

Summary of operation counts to solve LS problems, to evaluate the estimate $\widetilde{\mu}(x)$, and to evaluate the exact $\mu(x)$. Only leading terms are considered.

task	operations	$m = 1000$, $n = 100$	source
solve LS by QR	$2mn^2$	20,000,000	Table 4.2
solve LS by QR and evaluate $\widetilde{\mu}(x)$ by (4.3)	$2mn^2 + \frac{8}{3}n^3$	22,666,667	Table 4.2
solve LS by Chan SVD	$2mn^2 + 11n^3$	31,000,000	Table 4.1
solve LS by Chan SVD and evaluate $\widetilde{\mu}(x)$ by (4.3)	$2mn^2 + \frac{41}{3}n^3$	33,666,667	Tables 4.1, 4.2
solve LS by Chan SVD and evaluate $\widetilde{\mu}(x)$ by (4.1)	$6mn^2 + 20n^3$	80,000,000	Table 4.1
evaluate $\mu(x)$ by (1.1)	$4m^3 + 2m^2n$	4,200,000,000	[9, p. 239]

This formula could use two *reduced* QR factorizations. Of course, $Y_{K'}$ needn't be stored because $Y_{K'}^t v'$ can be accumulated as K' is reduced to triangular form.

Table 4.2 shows that the optimal backward error can be estimated at little additional cost over that of solving the LS problem when $m \gg n$. Since K' is a $2n \times n$ matrix, its QR factorization needs only $\mathcal{O}(n^3)$ operations compared to $\mathcal{O}(mn^2)$ for the factorization of A. Karlson and Waldén [15, p. 864] considered this same calculation in the course of evaluating a different estimate for the optimal backward error. They noted that sweeps of plane rotations most economically eliminate the lower block of K' while retaining the triangular structure of R_A.

4.3. Operation counts for dense matrix methods. Table 4.3 summarizes the operation counts of solving the LS problem and estimating its optimal backward errors by the QR and SVD solution methods for dense matrices. It is clear that evaluating the estimate is negligible compared to evaluating the true optimal backward error. *Obtaining the estimate is even negligible compared to solving the LS problem by QR methods.*

The table shows that the QR approach also gives the most effective way to evaluate $\widetilde{\mu}(x)$ when the LS problem is solved by SVD methods. Chan's algorithm for calculating the SVD begins by performing a QR factorization. Saving this intermediate factorization allows (4.3) to evaluate the estimate with the same, small marginal cost as in the purely QR case of Table 4.3.

4.4. Sparse QR methods. Equation (4.3) uses both factors of A's QR decomposition: Y_A to transform r, and R_A occurs in K'. Although progress has been made

towards computing both QR factors of a sparse matrix, notably by Adlers [1], it is considerably easier to work with just the triangular factor, as described by Matstoms [17]. Therefore methods to evaluate $\widetilde{\mu}(x)$ are needed that do not presume Y_A.

The simplest approach may be to evaluate (3.6) directly by transforming K to upper triangular form. Notice that A^tA and K^tK have identical sparsity patterns. Thus the same elimination analysis would serve to determine the sparse storage space for both R_A and R. Also, Y^tv can be obtained from QR factors of $\begin{bmatrix} K & v \end{bmatrix}$. The following MATLAB code is often effective for computing $\widetilde{\mu}(x)$ for a sparse matrix A and a dense vector b:

```
[m,n] = size(A);            r    = b - A*x;
normx = norm(x);            eta  = norm(r)/normx;
p     = colamd(A);
K     = [A(:,p);  eta*speye(n)];
v     = [    r ;    zeros(n,1)];
[c,R] = qr(K,v,0);          muKW = norm(c)/normx;
```

Note that `colamd` returns a good permutation `p` without forming `A'*A`, and `qr(K,v,0)` computes the required projection `c` $= Y^tv$ without storing any Q.

4.5. Iterative methods. If A is too large to permit the use of direct methods, we may consider iterative solution of the original problem $\min \|Ax-b\|$ as well as the KW problem (3.1):

$$\min_y \|Ky - v\| \;\equiv\; \min_y \left\| \begin{bmatrix} A \\ \eta I \end{bmatrix} y - \begin{bmatrix} r \\ 0 \end{bmatrix} \right\|, \qquad \eta \equiv \eta(x) = \frac{\|r\|}{\|x\|}. \tag{4.4}$$

In particular, LSQR [20, 21, 23] takes advantage of the damped least squares structure in (4.4). Using results from Saunders [22], we show here that the required projection norm is available within LSQR at negligible additional cost.

For problem (4.4), LSQR uses the Golub-Kahan bidiagonalization of A to form matrices U_k and V_k with theoretically orthonormal columns and a lower bidiagonal matrix B_k at each step k. With $\beta_1 = \|r\|$, a damped LS subproblem is defined and transformed by a QR factorization:

$$\min_{w_k} \left\| \begin{bmatrix} B_k \\ \eta I \end{bmatrix} w_k - \begin{bmatrix} \beta_1 e_1 \\ 0 \end{bmatrix} \right\|, \qquad Q_k \begin{bmatrix} B_k & \beta_1 e_1 \\ \eta I & 0 \end{bmatrix} = \begin{bmatrix} R_k & z_k \\ & \bar{\zeta}_{k+1} \\ & q_k \end{bmatrix}. \tag{4.5}$$

The kth estimate of y is defined to be $y_k = V_k w_k = (V_k R_k^{-1}) z_k$. From [22, pp. 99–100], the kth estimate of the required projection is given by

$$Ky \approx Ky_k \equiv \begin{bmatrix} A \\ \eta I \end{bmatrix} y_k \;=\; \begin{bmatrix} U_{k+1} & \\ & V_k \end{bmatrix} Q_k^t \begin{bmatrix} z_k \\ 0 \end{bmatrix}. \tag{4.6}$$

Orthogonality (and exact arithmetic) gives $\|Ky_k\| = \|z_k\|$. Thus if LSQR terminates at iteration k, $\|z_k\|$ may be taken as the final estimate of $\|Ky\|$ for use in (3.6), giving $\widetilde{\mu}(x) \approx \|z_k\|/\|x\|$. Since z_k differs from z_{k-1} only in its last element, only k operations are needed to accumulate $\|z_k\|^2$.

LSQR already forms monotonic estimates of $\|y\|$ and $\|v - Ky\|$ for use in its stopping rules, and the estimates are returned as output parameters. We see that the estimate $\|z_k\| \approx \|Ky\|$ is another useful output. Experience shows that the estimates of such norms retain excellent accuracy even though LSQR does not use reorthogonalization.

TABLE 5.1
Matrices used in the numerical tests.

matrix	rows m	columns n	$\kappa_2(A)$
(a) `illc1033`	1033	320	$1.9e{+}4$
(b) `well1033`	1033	320	$1.7e{+}2$
(c) `prolate`	$100 \le m \le 1000$	$n < m$	up to $9.2e{+}10$

5. Numerical tests with direct methods. This section presents numerical tests of the optimal backward error estimate. For this purpose it is most desirable to make many tests with problems that occur in practice. Since large collections of test problems are not available for least squares, it is necessary to compromise by using many randomly generated vectors, b, with a few matrices, A, that are related to real-world problems.

5.1. Description of the test problems. The first two matrices in Table 5.1, `well1033` and `illc1033`, originated in the least-squares analysis of gravity-meter observations. They are available from the Harwell-Boeing sparse matrix collection [7] and the Matrix Market [4]. The `prolate` matrices [27] are very ill-conditioned Toeplitz matrices of a kind that occur in signal processing. As generated here their entries are parameterized by a number a:

$$A_{i,j} = \begin{cases} 2a & \text{if } i = j, \\ \dfrac{\sin(2a\pi|i-j|)}{\pi|i-j|} & \text{if } i \neq j. \end{cases}$$

Since these matrices may be given any dimensions, a random collection is uniformly generated with $100 \le m \le 1000$, $1 \le n < m$, and $-\frac{1}{4} \le a \le \frac{1}{4}$.

Without loss of generality the vectors b in the LS problems may be restricted to norm 1. Sampling them uniformly from the unit sphere poses a subtle problem because, if A has more rows than columns, most vectors on the unit sphere are nearly orthogonal to range(A). (To see this in 3-space, suppose that range(A) is the earth's axis. The area below 45° latitude is much larger than the surface area in higher latitudes. This effect is more pronounced for higher dimensional spheres.) Since least squares problems are only interesting when b has some approximation in terms of A's columns, b is sampled so that $\angle(b, \text{range}(A))$ is uniformly distributed, as follows:

$$b \;=\; c_1 \frac{P_A u}{\|P_A u\|} + c_2 \frac{(I - P_A)u}{\|(I - P_A)u\|}.$$

In this formula, $(c_1, c_2) = (\cos(\theta), \sin(\theta))$ is uniformly chosen on the unit sphere, $P_A : \mathbb{R}^m \to \text{range}(A)$ is the orthogonal projection, and u is uniformly chosen on the unit sphere in $\mathbb{R}^m$ using the method of Calafiore, Dabbene, and Tempo [5].

5.2. Description of the calculations. For the factorization methods, 1000 sample problems are considered for each type of matrix in Table 5.1. For each sample problem, the solution x and the backward error estimate $\widetilde{\mu}(x)$ are computed using IEEE single precision arithmetic. The estimates are compared with the optimal backward error $\mu(x)$ from Higham's equation (1.1) evaluated in double precision. Matrix decompositions are calculated and manipulated using the LINPACK [6] subroutines `sqrdc`, `sqrsl`, and `ssvdc`, and for the higher precision calculations `dqrdc`, `dqrsl`, and `dsvdc`.

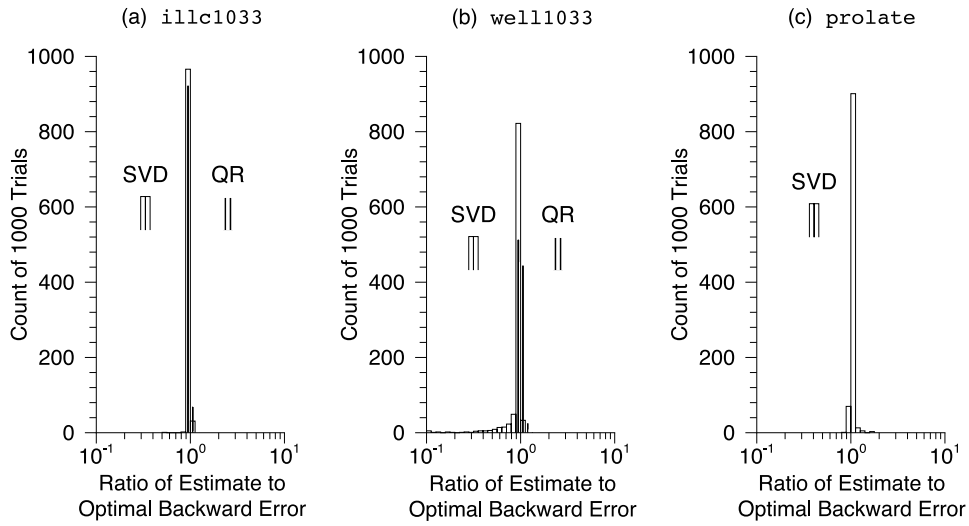

FIG. 5.1. *Histograms for the ratios of single precision estimates to true optimal backward error for all the test cases solved by dense matrix factorizations. The SVD and QR solution methods use the estimates in (4.1) and (4.3), respectively.*

Single precision is used for the tests so it is possible to assess the rounding error in the estimates by comparing them to the values formed with greater precision. It should be understood that the errors in the solutions and in the estimates are exaggerated as a result; they are larger than would be obtained by calculating solutions to the same problems with double precision arithmetic.

5.3. Test results for SVD and QR methods. Figure 5.1 displays the ratios

$$\widetilde{\mu}(x)|_{\text{single}} \,/\, \mu(x) \tag{5.1}$$

for all the test cases. The notation $|_{\text{single}}$ is used to emphasize that the estimate $\widetilde{\mu}(x)$ itself must be obtained by machine arithmetic, in this case by single precision arithmetic. Figure 5.1 shows that in most cases $\widetilde{\mu}(x)|_{\text{single}}$ is indistinguishable from the optimal backward error. The remainder of this section is a detailed examination of the test results for each problem set.

Tests with (a) illc1033. Figure 5.2 displays more information about the tests for the `illc1033` matrix. Since this matrix has full rank, the LS problem has a unique solution x_*, so it is meaningful to consider the relative solution error,

$$\|x - x_*\| \,/\, \|x_*\| \,. \tag{5.2}$$

This is displayed in Figure 5.2(a). *It is surprising that the SVD solution method produces larger errors than the QR method.* Nevertheless, Figure 5.2(b) shows that in all cases the relative backward error,

$$\mu(x) \,/\, \|A\|_{\text{F}} \,, \tag{5.3}$$

is smaller than the single precision rounding unit. *Another surprise is that the backward errors of LS problems are orders of magnitude smaller than the solution errors.* Thus, by von Neumann's criterion, all the x's computed for the sample problems must be accepted as accurate solutions.

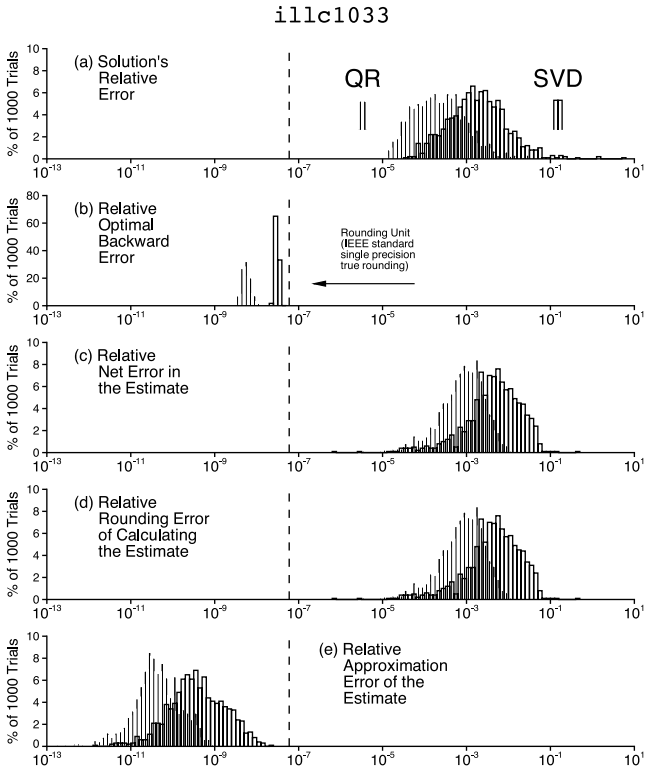

FIG. 5.2. *For the* `illc1033` *test cases, these figures show the relative size of: (a) solution error, (b) optimal backward error, (c, d, e) errors in the estimated optimal backward error. These quantities are defined in (5.2), (5.3), (5.4), (5.6), and (5.7), respectively.*

The scatter away from 10^0 in Figure 5.1's ratios is explained by the final three parts of Figure 5.2. The overall discrepancy in $\widetilde{\mu}(x)|_{\text{single}}$ as compared to $\mu(x)$ is shown in Figure 5.2(c):

$$\left| \, \widetilde{\mu}(x)|_{\text{single}} - \mu(x) \, \right| / \, \mu(x) \; . \tag{5.4}$$

This discrepancy is the sum of rounding error and approximation error:

$$\widetilde{\mu}(x)|_{\text{single}} - \mu(x) \;=\; \underbrace{\left[\; \widetilde{\mu}(x)|_{\text{single}} - \widetilde{\mu}(x) \; \right]}_{\text{rounding error}} \;+\; \underbrace{\left[\; \widetilde{\mu}(x) - \mu(x) \; \right]}_{\text{approximation error}} \; . \tag{5.5}$$

Figure 5.2(d) shows the relative rounding error,

$$\left| \, \widetilde{\mu}(x)|_{\text{single}} - \widetilde{\mu}(x) \, \right| / \, \widetilde{\mu}(x) \; , \tag{5.6}$$

while Figure 5.2(e) shows the relative approximation error,

$$\left| \, \widetilde{\mu}(x) - \mu(x) \, \right| / \, \mu(x) \; . \tag{5.7}$$

Yet another surprise from Figure 5.2(e) is that the approximation error of the estimate is vanishingly small. Evidently (1.6)'s limit approaches 1 so quickly that the approximation error is by far the smaller quantity in (5.5). *Thus the scatter in Figure 5.1's ratios is due primarily to rounding errors in evaluating the estimate.*

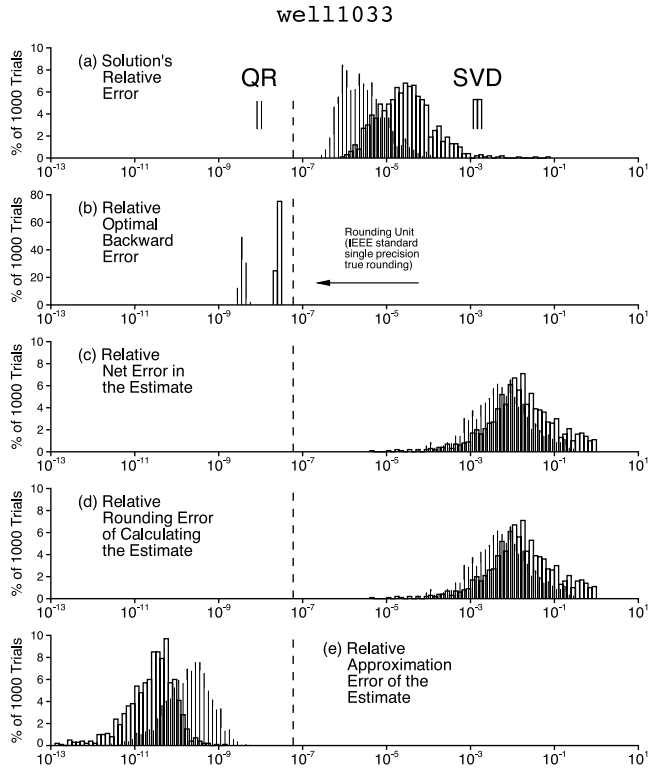

FIG. 5.3. *For the* `well1033` *test cases, these figures show the relative size of: (a) solution error, (b) optimal backward error, (c, d, e) errors in the estimated optimal backward error. These quantities are defined in (5.2), (5.3), (5.4), (5.6), and (5.7), respectively.*

As a result of this scatter, it should be pointed out, *the computed estimate often does not satisfy Gu's lower bound in (1.5),*

$$1 \approx \frac{\|r_*\|}{\|r\|} \not\leq \frac{\widetilde{\mu}(x)\big|_{\text{single}}}{\mu(x)} .$$

The bound fails even when x and $\widetilde{\mu}(x)$ are evaluated in double precision. For higher precisions the scatter does decrease, but the lower bound becomes more stringent because r becomes a better approximation to r_*.

Tests with (b) well1033. Figure 5.3 shows the details of the tests for the better conditioned matrix `well1033`. Some differences between it and Figure 5.2 can be partially explained by the conditioning of LS problems. The relative, spectral-norm condition number of the full-rank LS problem is [3, p. 31, eqn. 1.4.28] [11, thm. 5.1]

$$\left(\frac{\|r_*\|}{\sigma_{\min}(A)\,\|x_*\|} + 1 \right) \kappa_2(A) . \tag{5.8}$$

As A becomes better conditioned so does the LS problem, and the calculated solutions should become more accurate. This is the trend in the leftward shift of the histograms from Figure 5.2(a) to 5.3(a).

However, the rightward shift from Figure 5.2(d) to 5.3(d) suggests that *as x becomes more accurate, $\widetilde{\mu}(x)$ may become more difficult to evaluate accurately.* The

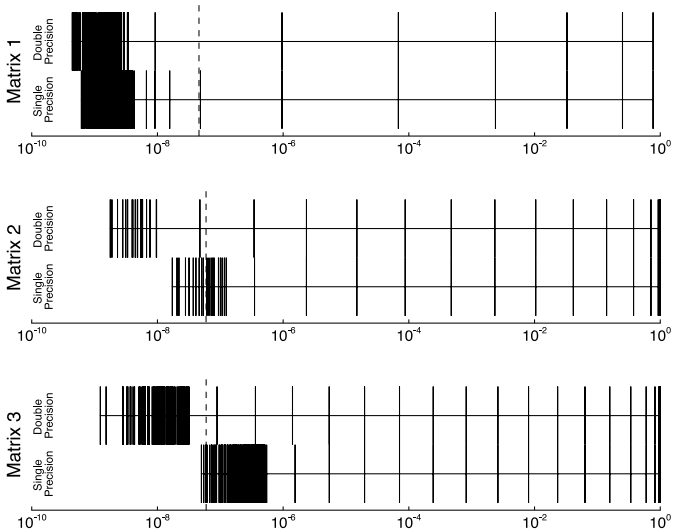

FIG. 5.4. *True and computed (that is, double and single precision) singular values for three of the* `prolate` *test matrices. Both calculations begin from the same data because the matrices are generated in single precision and then are promoted to double precision. The smallest single precision values are wildly inaccurate even above the cutoff (dashed line) for numerical rank.*

reason is that r, too, is more accurate, so v is more nearly orthogonal to range(K) in (3.6). With $\|Pv\|$ smaller, the rounding error in $r|_{\text{single}}$ accounts for more of the projection's value. Just the error of rounding r to single precision places a floor on the error in any machine representation, so this error never can be eliminated entirely.

Not too much should be inferred from the reversed position of the QR and SVD data in Figures 5.2(e) and 5.3(e). These figures are subject to rounding error because they actually compare double precision evaluations of $\widetilde{\mu}(x)$ and $\mu(x)$. The small error in these quantities cannot be checked without comparing them to quad precision calculations, which are very slow.

Tests with (c) prolate matrices. Table 5.1 indicates that the prolate matrices are rank-deficient in single precision. In this case, the default approach taken by LAPACK's driver routines [2, pp. 12, 141] is to truncate the SVD. This replaces A by $U_1\Sigma_1V_1^t$, where Σ_1 consists of all singular values $\sigma_i \geq \mathbf{u}\,\sigma_{\max}(A)$ and where $\mathbf{u}$ is the roundoff unit. In statistics, the effect is to reduce the variance in the estimator, x, at the cost of introducing some bias [3, p. 100]. In numerical terms, the truncated problem has the advantages of being well posed and avoiding large magnitudes in the solution that might be caused by small divisors. The numerical justification for changing the problem is that the truncated SVD perturbs A by no more than what is needed to represent A in finite precision. Discussions of this approach in the literature usually are not accompanied by examples such as Figure 5.4, which displays the true and computed singular values for three test matrices. From the figure it is not clear that the truncated approach is meaningful because the smallest singular values, which determine the truncation, appear to be computed with no accuracy whatsoever.

Figure 5.5 displays details of the `prolate` test cases for solutions computed by the truncated SVD. The solution's relative error in Figure 5.5(a) is with respect to the unique solution of the LS problem, which does have full rank in higher precision.

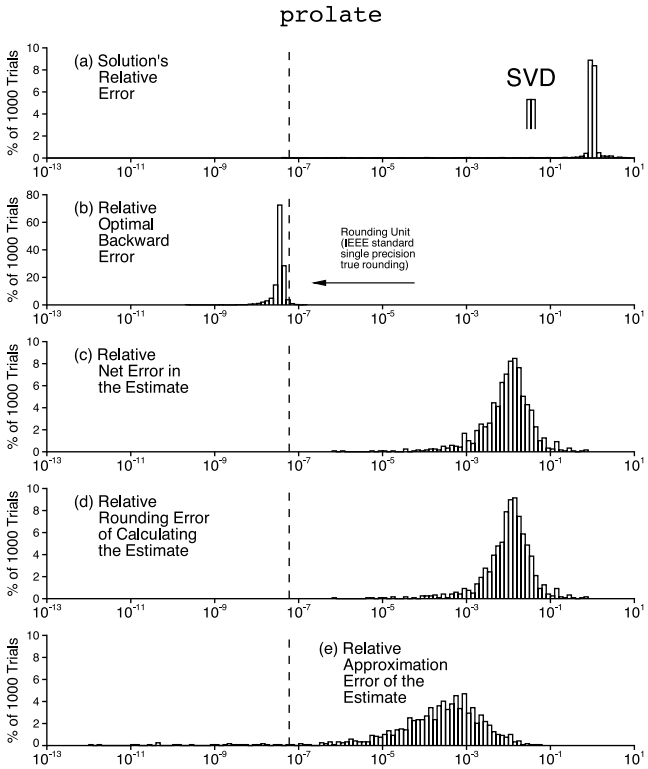

FIG. 5.5. *For the* `prolate` *test cases, these figures show the relative size of: (a) solution error, (b) optimal backward error, (c, d, e) errors in the estimated optimal backward error. These quantities are defined in (5.2), (5.3), (5.4), (5.6), and (5.7), respectively.*

Although the single precision solutions are quite different from x_*, Figure 5.5(b) indicates that the backward errors are acceptably small. The small backward errors *do* justify using the truncated SVD to solve these problems. This suggests that the ability to estimate the backward error might be useful in designing algorithms for rank-deficient problems. For example, in the absence of a problem-dependent criterion, small singular values might be included in the truncated SVD provided they do not increase the backward error.

The rest of Figure 5.5 is consistent with the other tests. The rounding error in the estimate has about the same relative magnitude, $\mathcal{O}(10^{-2})$, in all Figures 5.2(d), 5.3(d), and 5.5(d). The approximation error shown in Figure 5.5(e) is larger than in Figures 5.2(e) and 5.3(e) because of the much less accurate solutions, but overall, $\widetilde{\mu}(x)|_{\text{single}}$ remains an acceptable backward error estimate. Indeed, the estimate is remarkably good given the poor quality of the computed solutions.

6. Test results for iterative methods. In the preceding numerical results, the vector x has been an accurate estimate of the LS solution. Applying LSQR to a problem $\min_x \|Ax - b\|$ generates a sequence of *approximate* solutions $\{x_k\}$. For the `well` and `illc` test problems we used the MATLAB code in section 4.4 to compute $\widetilde{\mu}(x_k)$ for each x_k.

To our surprise, these values proved to be monotonically decreasing, as illustrated by the lower curve in Figures 6.1 and 6.2. (To make it scale-independent, this curve

is really $\widetilde{\mu}(x_k)/\|A\|_{\mathrm{F}}$.)

For each x_k, let $r_k = b - Ax_k$ and $\eta(x_k) = \|r_k\|/\|x_k\|$. Also, let K_k, v_k and y_k be the quantities in (3.1) when $x = x_k$. The LSQR iterates have the property that $\|r_k\|$ and $\|x_k\|$ are decreasing and increasing respectively, so that $\eta(x_k)$ is monotonically decreasing. Also, we see from (3.6) that

$$\widetilde{\mu}(x_k) = \frac{\|Y_k^t v_k\|}{\|x_k\|} < \frac{\|v_k\|}{\|x_k\|} = \frac{\|r_k\|}{\|x_k\|} = \eta(x_k),$$

so that $\eta(x_k)$ forms a monotonically decreasing *bound* on $\widetilde{\mu}(x)$. However, we can only note empirically that $\widetilde{\mu}(x_k)$ itself appears to decrease monotonically also.

The stopping criterion for LSQR is of interest. It is based on a *non-optimal* backward error $\|E_k\|_{\mathrm{F}}$ derived by Stewart [24], where

$$E_k = -\frac{1}{\|r_k\|^2} r_k r_k^t A.$$

(If $\tilde{A} = A + E_k$ and $\tilde{r} = b - \tilde{A}x_k$, then $(x_k, \tilde{r}_k)$ are the exact solution and residual for $\min_x \|\tilde{A}x - b\|$.) Note that $\|E_k\|_{\mathrm{F}} = \|E_k\|_2 = \|A^t r_k\|/\|r_k\|$. On incompatible systems, LSQR terminates when its estimate of $\|E_k\|_2/\|A\|_{\mathrm{F}}$ is sufficiently small; i.e., when

$$\texttt{test2}_k \equiv \frac{\|A^t r_k\|}{\|A\|_k \|r_k\|} \;\le\; \texttt{atol}, \tag{6.1}$$

where $\|A\|_k$ is a monotonically increasing estimate of $\|A\|_{\mathrm{F}}$ and `atol` is a user-specified tolerance.

Figures 6.1 and 6.2 show $\|r_k\|$ and three relative backward error quantities for problems `well1033` and `illc1033` when LSQR is applied to $\min_x \|Ax - b\|$ with $\texttt{atol} = 10^{-12}$. From top to bottom, the curves plot the following ($\log_{10}$):

- $\|r_k\|$ (monotonically decreasing).
- $\texttt{test2}_k$, LSQR's relative backward error estimate (6.1).
- $\eta(x_k)/\|A\|_{\mathrm{F}}$, the optimal relative backward error for $Ax = b$ (monotonic).
- $\widetilde{\mu}(x_k)/\|A\|_{\mathrm{F}}$, the KW relative backward error estimate for $\min_x \|Ax - b\|$, where $\widetilde{\mu}(x_k) = \|Pv_k\|/\|x_k\|$ in (3.6) is evaluated as in section 4.4. (It is apparently monotonic.)

The last curve is extremely close to the *optimal* relative backward error for LS problems. We see that LSQR's $\texttt{test2}_k$ is two or three orders of magnitude larger for most x_k, and it is far from monotonic. Nevertheless, its trend is downward in broad synchrony with $\widetilde{\mu}(x_k)/\|A\|_{\mathrm{F}}$. We take this as an experimental approval of Stewart's backward error E_k and confirmation of the reliability of LSQR's cheaply computed stopping rule.

6.1. Iterative computation of $\widetilde{\mu}(x)$. Here we use an iterative solver twice: first on the original LS problem to obtain an approximate solution x, and then on the associated KW problem to estimate the backward error for x.

1. Apply LSQR to $\min_x \|Ax - b\|$ with iteration limit *kmax*. This generates a sequence $\{x_k\}$, $k = 1\!:\!kmax$. Define $x = x_{kmax}$. We want to estimate the backward error for that final point x.
2. Define $r = b - Ax$ and $\texttt{atol} = 0.01\|A^t r\|/(\|A\|_{\mathrm{F}}\|x\|)$.
3. Apply LSQR to the KW problem $\min_y \|Ky - v\|$ (4.4) with convergence tolerance `atol`. As described in section 4.5, this generates a sequence of estimates $\widetilde{\mu}(x) \approx \|z_\ell\|/\|x\|$ using $\|z_\ell\| \approx \|Ky\|$ in (4.5)–(4.6).

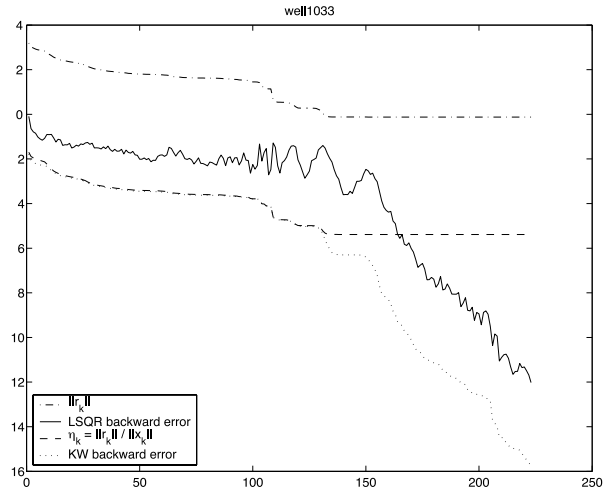

FIG. 6.1. *Backward error estimates for each LSQR iterate x_k during the solution of* `well1033` *with* `atol` $= 10^{-12}$. *The middle curve is Stewart's estimate as used in LSQR; see* (6.1). *The bottom curve is $\widetilde{\mu}(x_k)/\|A\|_F$, where $\widetilde{\mu}(x_k)$ is the KW bound on the optimal backward error computed as in section 4.4 (unexpectedly monotonic).*

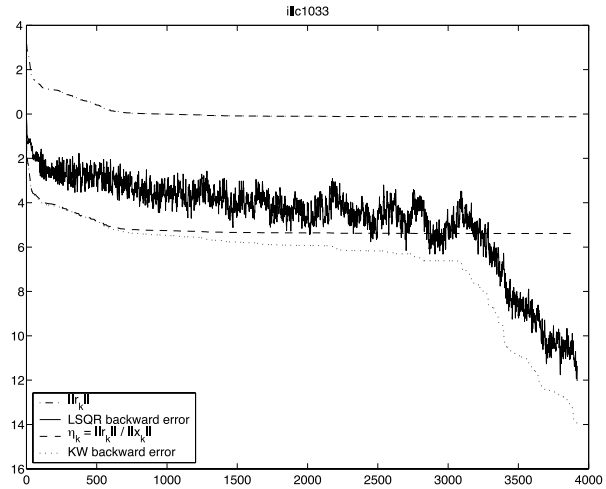

FIG. 6.2. *Backward error estimates for each LSQR iterate x_k during the solution of* `illc1033` *with* `atol` $= 10^{-12}$.

To avoid ambiguity we use k and ℓ for LSQR's iterates on the two problems. Also, the following figures plot *relative* backward errors $\widetilde{\mu}(x)/\|A\|_{\mathrm{F}}$, even though the accompanying discussion doesn't mention $\|A\|_{\mathrm{F}}$.

For problem `well1033` with $kmax = 50$, Figure 6.3 shows $\widetilde{\mu}(x_k)$ for $k = 1:50$ (the same as the beginning of Figure 6.1). The right-hand curve then shows about 130 estimates $\|z_\ell\|/\|x\|$ converging to $\widetilde{\mu}(x_{50})$ with about 2 digits of accuracy (because of the choice of `atol`).

Similarly with $kmax = 160$, Figure 6.4 shows $\widetilde{\mu}(x_k)$ for $k = 1:160$ (the same as the beginning of Figure 6.1). The final point x_{160} is close to the LS solution, and the subsequent KW problem converges more quickly. About 20 LSQR iterations give a

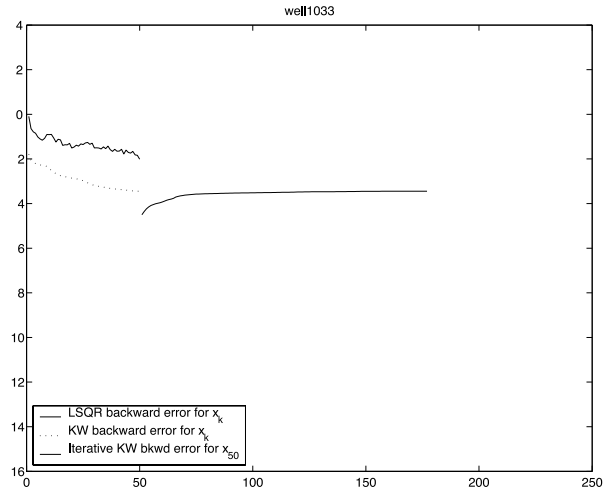

FIG. 6.3. *Problem* `well1033`*: Iterative solution of KW problem after LSQR is terminated at* x_{50}.

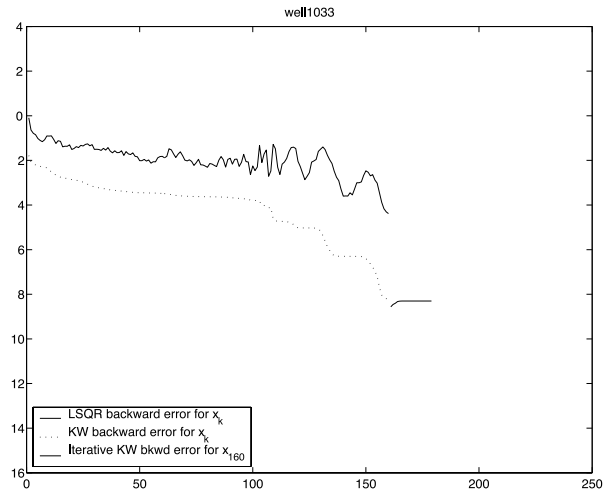

FIG. 6.4. *Problem* `well1033`*: Iterative solution of KW problem after LSQR is terminated at* x_{160}.

2-digit estimate of $\widetilde{\mu}(x_{160})$.

For problem `illc1033`, similar effects were observed. In Figure 6.5 about 2300 iterations on the KW problem give a 2-digit estimate of $\widetilde{\mu}(x_{2000})$, but in Figure 6.6 only 280 iterations are needed to estimate $\widetilde{\mu}(x_{3500})$.

6.2. Comparison with Malyshev and Sadkane's iterative method. Malyshev and Sadkane [16] show how to use the bidiagonalization of A with starting vector r to estimate $\sigma_{\min}[A\ \ B]$ in (1.1). This is the same bidiagonalization that LSQR uses on the KW problem (3.1) to estimate $\widetilde{\mu}(x)$. The additional work per iteration is nominal in both cases. A numerical comparison is therefore of interest. We use the results in Tables 5.2 and 5.3 of [16] corresponding to LSQR's iterates x_{50} and x_{160} on problems `well1033` and `illc1033`. Also, MATLAB gives us accurate values for $\widetilde{\mu}(x_k)$ and $\sigma_{\min}[A\ \ B]$ via sparse `qr` (section 4.4) and dense `svd` respectively.

In Tables 6.1–6.3, the true backward error is $\mu(x) = \sigma_{\min}[A\ \ B]$, the last line in

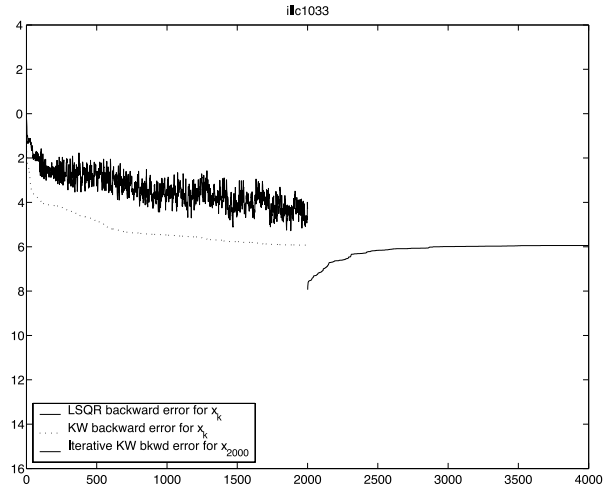

FIG. 6.5. *Problem* `illc1033`*: Iterative solution of KW problem after LSQR is terminated at* x_{2000}.

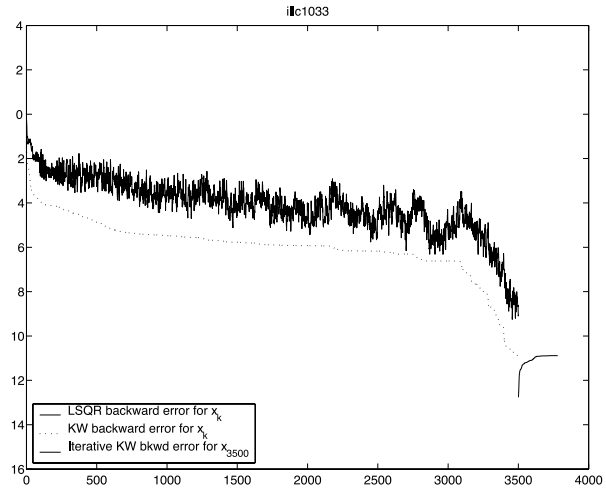

FIG. 6.6. *Problem* `illc1033`*: Iterative solution of KW problem after LSQR is terminated at* x_{3500}.

each table.

In Tables 6.1–6.2, σ_ℓ denotes Malyshev and Sadkane's $\sigma_{\min}(\bar{B}_\ell)$ [16, (3.7)]. Note that the iterates σ_ℓ provide *decreasing upper bounds on* $\sigma_{\min}[A\ B]$, while the LSQR iterates $\|z_\ell\|/\|x\|$ are *increasing lower bounds on* $\widetilde{\mu}(x)$, but they do not bound $\sigma_{\min}$.

We see that all of the Malyshev and Sadkane estimates σ_ℓ bound $\sigma_{\min}$ to within a factor of 2, but they have no significant digits in agreement with $\sigma_{\min}$. In contrast, $\eta(x_k)$ agrees with $\sigma_{\min}$ to 3 digits in three of the cases, and indeed it provides a tighter bound whenever it satisfies $\eta < \sigma_\ell$. The estimates σ_ℓ are therefore more valuable when $\eta > \sigma_{\min}$ (i.e., when x_k is close to a solution x_*).

However, we see that LSQR computes $\widetilde{\mu}(x_k)$ with 3 or 4 correct digits in all cases, and requires fewer iterations as x_k approaches x_*. The bottom-right values in Tables 6.1 and 6.3 show Grcar's limit (1.6) taking effect. LSQR can compute these values to high precision with reasonable efficiency.

TABLE 6.1
Comparison of σ_ℓ and $\|z_\ell\|/\|x_k\|$ for problem `well1033`.

$k = 50$

$\|r_k\|$	6.35e+1	
$\|A^t r_k\|$	5.04e+0	
$\eta(x_k)$		7.036807e−3
`atol`	4.44e−5	
ℓ	σ_ℓ	$\|z_\ell\|/\|x_k\|$
10	2.35e−2	2.11e−3
50	1.51e−2	5.43e−3
100	1.22e−2	6.32e−3
127		6.379461e−3
$\widetilde{\mu}(x_k)$		6.379462e−3
$\sigma_{\min}[A\ \ B]$		7.036158e−3

$k = 160$

$\|r_k\|$	7.52e−1	
$\|A^t r_k\|$	4.49e−4	
$\eta(x_k)$		7.3175e−5
`atol`	3.34e−7	
ℓ	σ_ℓ	$\|z_\ell\|/\|x_k\|$
10	3.79e−5	8.9316e−8
19		8.9381e−8
50	2.95e−7	
100	1.21e−7	
$\widetilde{\mu}(x_k)$		8.9386422278e−8
$\sigma_{\min}[A\ \ B]$		8.9386422275e−8

TABLE 6.2
Comparison of σ_ℓ and $\|z_\ell\|/\|x_k\|$ for problem `illc1033`.

$k = 50$

$\|r_k\|$	3.67e+1	
$\|A^t r_k\|$	3.08e+1	
$\eta(x_k)$		4.6603e−3
`atol`	4.69e−5	
ℓ	σ_ℓ	$\|z_\ell\|/\|x_k\|$
10	3.04e−2	1.62e−3
50	1.84e−2	3.71e−3
100	1.02e−2	4.11e−3
200		4.25e−3
300		4.28e−3
310		4.2825e−3
$\widetilde{\mu}(x_k)$		4.2831e−3
$\sigma_{\min}[A\ \ B]$		4.6576e−3

$k = 160$

$\|r_k\|$	1.32e+1	
$\|A^t r_k\|$	3.78e−1	
$\eta(x_k)$		1.6196e−3
`atol`	1.60e−5	
ℓ	σ_ℓ	$\|z_\ell\|/\|x_k\|$
10	1.10e−2	2.09e−4
50	4.63e−3	4.92e−4
100	3.40e−3	8.45e−4
200		1.23e−3
300		1.34e−3
400		1.38e−3
500		1.3841e−3
542		1.3843e−3
$\widetilde{\mu}(x_k)$		1.3847e−3
$\sigma_{\min}[A\ \ B]$		1.6144e−3

TABLE 6.3
$\|z_\ell\|/\|x_k\|$ for problem `illc1033`.

$k = 2000$

$\|r_k\|$	7.89e−1
$\|A^t r_k\|$	2.45e−3
$\eta(x_k)$	7.82e−5
`atol`	1.73e−6
ℓ	$\|z_\ell\|/\|x_k\|$
500	1.22e−5
1000	1.81e−5
1500	1.97e−5
2000	2.02e−5
2330	2.08e−5
$\widetilde{\mu}(x_k)$	2.10e−5
$\sigma_{\min}[A\ \ B]$	2.12e−5

$k = 3500$

$\|r_k\|$	7.52e−1
$\|A^t r_k\|$	5.54e−8
$\eta(x_k)$	7.30e−5
`atol`	4.11e−11
ℓ	$\|z_\ell\|/\|x_k\|$
10	4.41e−11
50	1.11e−10
100	1.54e−10
200	2.28e−10
280	2.32006e−10
$\widetilde{\mu}(x_k)$	2.3209779030e−10
$\sigma_{\min}[A\ \ B]$	2.3209779099e−10

The primary difficulty with our iterative computation of $\widetilde{\mu}(x)$ is that when x is *not* close to x_*, rather many iterations may be required, and there is no warning that $\widetilde{\mu}$ may be an underestimate of μ.

Ironically, solving the KW problem for $x = x_k$ is akin to restarting LSQR on a slightly modified problem. We have observed that if ℓ iterations are needed on the KW problem to estimate $\widetilde{\mu}(x_k)/\|A\|_{\mathrm{F}}$, continuing the original LS problem a further ℓ iterations would have given a point $x_{k+\ell}$ for which the Stewart-type backward error $\texttt{test2}_{k+\ell}$ is generally at least as small. (Compare Figures 6.2 and 6.6.) Thus, the decision to estimate *optimal* backward errors by iterative means must depend on the real need for optimality.

7. Conclusions. Several approaches have been suggested and tested to evaluate an estimate for the optimal size (that is, the minimal Frobenius norm) of backward errors for LS problems. Specifically, to estimate the true backward error $\mu(x)$ in (1.1) (for an arbitrary vector x), we have studied the estimate $\widetilde{\mu}(x)$ in (1.3). The numerical tests support various conclusions as follows.

Regarding LS problems themselves:

1. The QR solution method results in noticeably smaller solution errors than the SVD method.
2. The optimal backward errors for LS problems are much smaller—often orders of magnitude smaller—than the solution errors.

Regarding the estimates:

3. The computed estimate of the optimal backward error is very near the true optimal backward error in all but a small percent of the tests.
   (a) Grcar's limit (1.6) for the ratio of the estimate to the optimal backward error appears to approach 1 very quickly.
   (b) The greater part of the fluctuation in the estimate is caused by rounding error in its evaluation.
4. Gu's lower bound (1.5) for the ratio of the estimate to the optimal backward error often fails in practice because of rounding error in evaluating the estimate.
5. As the computed solution of the LS problem becomes more accurate, the estimate may become more difficult to evaluate accurately because of the unavoidable rounding error in forming the residual.
6. For QR methods, evaluating the estimate is insignificant compared to the cost of solving a dense LS problem.
7. When iterative methods become necessary, applying LSQR to the KW problem is a practical alternative to the bidiagonalization approach of Malyshev and Sadkane [16], particularly when x is close to x_*. No special coding is required (except a few new lines in LSQR to compute $\|z_k\| \approx Ky$ as in section 4.5), and LSQR's normal stopping rules ensure at least some good digits in the computed $\widetilde{\mu}(x)$.
8. The smooth lower curves in Figures 6.1 and 6.2 suggest that when LSQR is applied to an LS problem, the optimal backward errors for the sequence of approximate solutions $\{x_k\}$ are (unexpectedly) monotonically decreasing.
9. The Stewart backward error used in LSQR's stopping rule (6.1) can be some orders of magnitude larger than the optimal backward error, but it appears to track the optimal error well.

REFERENCES

[1] M. Adlers. *Topics in Sparse Least Squares Problems*. PhD thesis, Linköping University, Sweden, 2000.

[2] 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*. Society for Industrial and Applied Mathematics, Philadelphia, 1992.

[3] A. Björck. *Numerical Methods for Least Squares Problems*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 1996.

[4] R. F. Boisvert, R. Pozo, K. Remington, R. Barrett, and J. Dongarra. The Matrix Market: a web repository for test matrix data. In R. F. Boisvert, editor, *The Quality of Numerical Software, Assessment and Enhancement*, pages 125–137. Chapman & Hall, London, 1997. The web address of the Matrix Market is `http://math.nist.gov/MatrixMarket/`.

[5] G. Calafiore, F. Dabbene, and R. Tempo. Radial and uniform distributions in vector and matrix spaces for probabilistic robustness. In D. E. Miller et al., editor, *Topics in Control and Its Applications*, pages 17–31. Springer, 2000. Papers from a workshop held in Toronto, Canada, June 29–30, 1998.

[6] J. J. Dongarra, J. R. Bunch, C. B. Moler, and G. W. Stewart. *LINPACK Users' Guide*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 1979.

[7] I. S. Duff, R. G. Grimes, and J. G. Lewis. Sparse matrix test problems. *ACM Transactions on Mathematical Software*, 15(1):1–14, 1989.

[8] G. H. Golub and W. Kahan. Calculating the singular values and pseudoinverse of a matrix. *SIAM Journal on Numerical Analysis*, 2:205–224, 1965.

[9] G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, second edition, 1989.

[10] J. F. Grcar. Differential equivalence classes for metric projections and optimal backward errors. Technical Report LBNL-51940, Lawrence Berkeley National Laboratory, 2002. Submitted for publication.

[11] J. F. Grcar. Optimal sensitivity analysis of linear least squares. Technical Report LBNL-52434, Lawrence Berkeley National Laboratory, 2003. Submitted for publication.

[12] M. Gu. Backward perturbation bounds for linear least squares problems. *SIAM Journal on Matrix Analysis and Applications*, 20(2):363–372, 1999.

[13] M. Gu. New fast algorithms for structured linear least squares problems. *SIAM Journal on Matrix Analysis and Applications*, 20(1):244–269, 1999.

[14] N. J. Higham. *Accuracy and Stability of Numerical Algorithms*. Society for Industrial and Applied Mathematics, Philadelphia, second edition, 2002.

[15] R. Karlson and B. Waldén. Estimation of optimal backward perturbation bounds for the linear least squares problem. *BIT*, 37(4):862–869, December 1997.

[16] A. N. Malyshev and M. Sadkane. Computation of optimal backward perturbation bounds for large sparse linear least squares problems. *BIT*, 41(4):739–747, December 2002.

[17] P. Matstoms. Sparse QR factorization in MATLAB. *ACM Trans. Math. Software*, 20:136–159, 1994.

[18] J. von Neumann and H. H. Goldstine. Numerical inverting of matrices of high order. *Bulletin of the American Mathematical Society*, 53(11):1021–1099, November 1947. Reprinted in [26, v. 5, pp. 479–557].

[19] W. Oettli and W. Prager. Compatibility of approximate solution of linear equations with given error bounds for coefficients and right-hand sides. *Num. Math.*, 6:405–409, 1964.

[20] C. C. Paige and M. A. Saunders. LSQR: An algorithm for sparse linear equations and sparse least squares. *ACM Trans. Math. Software*, 8(1):43–71, 1982.

[21] C. C. Paige and M. A. Saunders. Algorithm 583; LSQR: Sparse linear equations and least-squares problems. *ACM Trans. Math. Software*, 8(2):195–209, 1982.

[22] M. A. Saunders. Computing projections with LSQR. *BIT*, 37(1):96–104, 1997.

[23] M. A. Saunders. lsqr.f, lsqr.m. http://www.stanford.edu/group/SOL/software/lsqr.html, 2002.

[24] G. W. Stewart. Research development and LINPACK. In J. R. Rice, editor, *Mathematical Software III*, pages 1–14. Academic Press, New York, 1977.

[25] Z. Su. *Computational Methods for Least Squares Problems and Clinical Trials*. PhD thesis, Stanford University, 2005.

[26] A. H. Taub, editor. *John von Neumann Collected Works*. Macmillan, New York, 1963.

[27] J. M. Varah. The prolate matrix. *Linear Algebra Applicat.*, 187:269–278, 1993.

[28] B. Waldén, R. Karlson, and J.-G. Sun. Optimal backward perturbation bounds for the linear least squares problem. *Numerical Linear Algebra with Applications*, 2(3):271–286, 1995.