REGULARIZATION PARAMETER DETERMINATION FOR DISCRETE ILL-POSED PROBLEMS*

M. E. HOCHSTENBACH†, L. REICHEL‡, AND G. RODRIGUEZ§

Abstract. Straightforward solution of discrete ill-posed linear systems of equations or least-squares problems with error contaminated data does not, in general, give meaningful results, because propagated error destroys the computed solution. The problems have to be modified to reduce their sensitivity to the error in the data. The amount of modification is determined by a regularization parameter. It can be difficult to determine a suitable value of the regularization parameter when no knowledge of the norm of error in the data is available. This paper proposes a new simple technique for determining a value of the regularization parameter that can be applied in this situation. It is based on comparing computed solutions determined by Tikhonov regularization and truncated singular value decomposition. Analogous comparisons are proposed for large-scale problems. The technique for determining the regularization parameter implicity provides an estimate for the norm of the error in the data.

Key words. Ill-posed problem, regularization, noise level estimation, TSVD, Golub–Kahan bidiagonalization, Tikhonov regularization, heuristic parameter choice rule.

AMS subject classifications. 65F10, 65F22, 65R30.

1. Introduction.

We are concerned with the solution of least-squares problems

$$\min_{\mathbf{x}\in\mathbb{R}^n} \|A\mathbf{x}-\mathbf{b}\| \tag{1.1}$$

with a matrix $A \in \mathbb{R}^{m\times n}$ whose singular values "cluster" at the origin. Thus, A is severely ill-conditioned and may be singular. Least-squares problems of this kind are commonly referred to as discrete ill-posed problems. They arise from the discretization of ill-posed problems, but may also originate in discrete form, for instance, in image deblurring problems. For notational convenience, we will assume that $m \geq n$, but this restriction easily can be removed. Throughout this paper $\|\cdot\|$ denotes the Euclidean vector norm or the associated induced matrix norm.

The vector $\mathbf{b} \in \mathbb{R}^m$ in (1.1) represents available data and is assumed to be contaminated by white Gaussian noise $\mathbf{e} \in \mathbb{R}^m$ caused by measurement inaccuracies. Thus,

$$\mathbf{b} = \widehat{\mathbf{b}} + \mathbf{e}, \tag{1.2}$$

where $\widehat{\mathbf{b}}$ denotes the unknown error-free vector associated with $\mathbf{b}$. We refer to the relative error $\nu = \|\mathbf{e}\|/\|\widehat{\mathbf{b}}\|$ as the noise level.

Introduce the (unknown) error-free least-squares problem associated with (1.1),

$$\min_{\mathbf{x}\in\mathbb{R}^n} \|A\mathbf{x}-\widehat{\mathbf{b}}\|. \tag{1.3}$$

---

*Version June 1, 2014.

†Department of Mathematics and Computer Science, Eindhoven University of Technology, PO Box 513, 5600 MB, The Netherlands, www.win.tue.nl/∼hochsten. Research supported in part by an NWO Vidi grant.

‡Department of Mathematical Sciences, Kent State University, Kent, OH 44242, USA. E-mail: reichel@math.kent.edu. Research supported in part by NSF grant DMS-1115385.

§Dipartimento di Matematica e Informatica, Università di Cagliari, viale Merello 92, 09123 Cagliari, Italy. E-mail: rodriguez@unica.it.

We are interested in computing an accurate approximation of the solution of minimal Euclidean norm of (1.3). It is given by $\widehat{\mathbf{x}} = A^\dagger \widehat{\mathbf{b}}$, where $A^\dagger$ denotes the Moore–Penrose pseudoinverse of A. Note that due to the error $\mathbf{e}$ in $\mathbf{b}$ and the severe ill-conditioning of A, the solution of minimal Euclidean norm of (1.1), given by

$$\mathbf{x} = A^\dagger \mathbf{b} = \widehat{\mathbf{x}} + A^\dagger \mathbf{e},$$

generally is not a meaningful approximation of $\widehat{\mathbf{x}}$ due to a large propagated error $A^\dagger \mathbf{e}$.

This difficulty is commonly remedied by replacing the least-squares problem (1.1) by a nearby problem, whose solution is less sensitive to the error $\mathbf{e}$, and solving the latter problem. This replacement is known as regularization. Commonly used regularization methods include Tikhonov regularization and truncated singular value decomposition (TSVD). The simplest form of Tikhonov regularization replaces the minimization problem (1.1) by the penalized least-squares problem

$$\min_{\mathbf{x}\in\mathbb{R}^n} \{\|A\mathbf{x} - \mathbf{b}\|^2 + \mu^2 \|\mathbf{x}\|^2\}, \tag{1.4}$$

where $\mu > 0$ is a regularization parameter. The solution of (1.4) is given by

$$\mathbf{x}_\mu = (A^T A + \mu^2 I)^{-1} A^T \mathbf{b}, \tag{1.5}$$

where the superscript ^{T} denotes transposition. The value of the regularization parameter μ determines how well $\mathbf{x}_\mu$ approximates $\widehat{\mathbf{x}}$ and how sensitive $\mathbf{x}_\mu$ is to the error $\mathbf{e}$ in the available data $\mathbf{b}$.

When an accurate estimate ε of the error norm $\|\mathbf{e}\|$ is available, the regularization parameter $\mu > 0$ often is chosen to satisfy the discrepancy principle, i.e., $\mu > 0$ is determined so that

$$\|A\mathbf{x}_\mu - \mathbf{b}\| = \tau\varepsilon, \tag{1.6}$$

where $\tau > 1$ is a user-specified constant independent of ε; see [10, 14] for discussions on this parameter choice rule.

We are interested in the situation when no accurate estimate of $\|\mathbf{e}\|$ is available. Regularization parameter choice rules for this situation are commonly referred to as heuristic, because they may fail in certain situations. A large number of heuristic parameter choice rules have been developed, because of the importance of being able to determine a suitable value of the regularization parameter μ when no accurate estimate of $\|\mathbf{e}\|$ is known. The most well-known of these rules include the L-curve criterion, generalized cross validation, error estimation methods based on extrapolation, the quasi-optimality principle, and Regińska's method; see [3, 5, 10, 14, 23, 24, 25] for discussions and comparisons of the performance of these and several other heuristic parameter choice rules and [19] for an analysis. This paper presents a new heuristic method for choosing the parameter and compares it with several available heuristic rules as well as with the discrepancy principle.

TSVD is another popular regularization method for the solution of discrete ill-posed problems of the form (1.1). Introduce the singular value decomposition (SVD)

$$A = U\Sigma V^T,$$

where the matrices

$$U = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n] \in \mathbb{R}^{m\times n}, \qquad V = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n] \in \mathbb{R}^{n\times n}$$

have orthonormal columns. The nontrivial entries of

$$\Sigma = \mathrm{diag}[\sigma_1, \sigma_2, \ldots, \sigma_n] \in \mathbb{R}^{n\times n}$$

are the singular values of A. They are ordered according to

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > \sigma_{r+1} = \cdots = \sigma_n = 0,$$

where the parameter r is the rank of A. Let the matrix $\Sigma_k \in \mathbb{R}^{n\times n}$ be obtained by setting the last $n-k$ diagonal entries $\sigma_{k+1}, \sigma_{k+2}, \ldots, \sigma_n$ of Σ to zero, and introduce the matrix

$$A_k = U\Sigma_k V^T$$

of rank at most k. The TSVD method determines the approximate solutions

$$\mathbf{x}_k = A_k^\dagger \mathbf{b} = \sum_{j=1}^{k} \frac{\mathbf{u}_j^T \mathbf{b}}{\sigma_j}\, \mathbf{v}_j, \qquad k = 1, 2, \ldots, r. \tag{1.7}$$

The parameter k is a regularization parameter. It is easy to see that the solution norm $\|\mathbf{x}_k\|$ is monotonically increasing with k, while the associated residual norm $\|\mathbf{b} - A\mathbf{x}_k\|$ is a monotonically decreasing function of k.

The discrepancy principle prescribes that $\widehat{\mathbf{x}}$ be approximated by the TSVD solution $\mathbf{x}_k$, where k is the smallest integer such that

$$\|A\mathbf{x}_k - \mathbf{b}\| \leq \tau\varepsilon; \tag{1.8}$$

cf. (1.6). An analysis of this parameter choice rule can be found in [10].

Let the TSVD solution $\mathbf{x}_k$ and the Tikhonov solution $\mathbf{x}_\mu$ satisfy the discrepancy principle. Then numerous numerical experiments suggest that $\mathbf{x}_k$ and $\mathbf{x}_\mu$ approximate the desired solution $\widehat{\mathbf{x}}$ roughly equally well; see also Varah [27] for a discussion on this. We will make use of this observation to estimate the norm of the noise $\mathbf{e}$ in situations when no accurate estimate is known. The computed estimate of $\|\mathbf{e}\|$ allows us to apply the discrepancy principle to determine an approximate solution of (1.1).

The remainder of this paper is organized as follows. Section 2 describes our approach to comparing approximate solutions given by the TSVD and Tikhonov methods. These comparisons implicitly provide an estimate of the noise level in $\mathbf{b}$. How to handle large-scale problems is discussed in Section 3. Numerical experiments are presented in Sections 4 and 5, and concluding remarks can be found in Section 6.

2. Comparison of Tikhonov and TSVD approximate solutions. We describe a procedure for comparing Tikhonov solutions (1.5) to TSVD solutions (1.7) and in this manner determine suitable regularization parameters for these methods. To every TSVD solution $\mathbf{x}_k$, we may associate a Tikhonov solution $\mathbf{x}_\mu$ as follows. Let

$$\rho_k = \|\mathbf{b} - A\mathbf{x}_k\| \tag{2.1}$$

be the residual norm corresponding to the TSVD solution $\mathbf{x}_k$. We will compute the Tikhonov solution that gives the same residual norm. The following result shows that this is possible for most problems.

PROPOSITION 2.1. *Introduce the function*

$$\varphi(\mu) = \|\mathbf{b} - A\mathbf{x}_\mu\|^2, \tag{2.2}$$

where $\mathbf{x}_\mu$ *is defined by* (1.5)*, and assume that* $A^T\mathbf{b} \neq \mathbf{0}$*. Then* φ *can be expressed as*

$$\varphi(\mu) = \mathbf{b}^T(\mu^{-2}AA^T + I)^{-2}\,\mathbf{b},$$

which shows that φ *is a strictly increasing function of* $\mu > 0$*. Moreover, the equation*

$$\varphi(\mu) = \gamma$$

has a unique solution μ*, such that* $0 < \mu < \infty$*, for any choice of* γ *that satisfies* $\|\mathbf{b}_0\|^2 < \gamma < \|\mathbf{b}\|^2$*, where* $\mathbf{b}_0$ *denotes the orthogonal projection of* $\mathbf{b}$ *onto the null space of* A^T*.*

Proof. The result follows by substituting the SVD of A into (2.2) and (1.5). □

COROLLARY 2.2. *For each* $1 \leq k < r$*, there is a Tikhonov regularization parameter* $\mu = \mu_k$ *such that*

$$\|\mathbf{b} - A\mathbf{x}_{\mu_k}\| = \rho_k, \tag{2.3}$$

where ρ_k *is defined by* (2.1)*.*

Proof. The corollary follows from Proposition 2.1. □

The value of the regularization parameter μ_k in (2.3) can be determined, e.g., by a Newton method. This amounts to computing the limit λ of the sequence

$$\lambda_{\ell+1} = \lambda_\ell + \frac{1}{2}\left[\sum_{j=1}^{n}\frac{(\mathbf{u}_j^T\mathbf{b})^2}{(\sigma_j^2\lambda_\ell + 1)^2} - \bar{\rho}\right] \cdot \left[\sum_{j=1}^{n}\frac{(\mathbf{u}_j^T\mathbf{b})^2\sigma_j^2}{(\sigma_j^2\lambda_\ell + 1)^3}\right]^{-1},$$

where $\bar{\rho} = \rho_k^2 - \|(I - UU^T)\mathbf{b}\|^2$. The value of the regularization parameter then is given by $\mu_k = \lambda^{-1/2}$. This form of Newton's method is designed to perform well also when the noise free least-squares problem (1.3) is inconsistent.

Let $k_{\min}$ be the integer for which the difference norm

$$\delta_k = \|\mathbf{x}_{\mu_k} - \mathbf{x}_k\| \tag{2.4}$$

has its first local minimum as k increases, starting with $k = 1$. We choose $\mu = \mu_{k_{\min}}$ as the Tikhonov regularization parameter. This choice or regularization parameter is in agreement with the discrepancy principle if the noise in $\mathbf{b}$ is of the order $\rho_{k_{\min}}$. We therefore may use $\rho_{k_{\min}}$ as an estimate for the norm of the noise in $\mathbf{b}$. Algorithm 1 below implements the determination of $\mu_{k_{\min}}$ and $\rho_{k_{\min}}$. The computations of the algorithm are simple and inexpensive when the SVD of A is available. For instance, the Tikhonov solution is given by

$$\mathbf{x}_\mu = \sum_{j=1}^{r}\frac{\sigma_j}{\sigma_j^2 + \mu^2}\,(\mathbf{u}_j^T\mathbf{b})\,\mathbf{v}_j.$$

The following discussion sheds some light on why choosing the index k as described may be appropriate. Assume for the moment that $\|\mathbf{x}_{\mu_k} - \widehat{\mathbf{x}}\| = \|\mathbf{x}_k - \widehat{\mathbf{x}}\|$. Then

$$\|\mathbf{x}_{\mu_k} - \mathbf{x}_k\| \leq \|\mathbf{x}_{\mu_k} - \widehat{\mathbf{x}}\| + \|\mathbf{x}_k - \widehat{\mathbf{x}}\| = 2\,\|\mathbf{x}_k - \widehat{\mathbf{x}}\|.$$

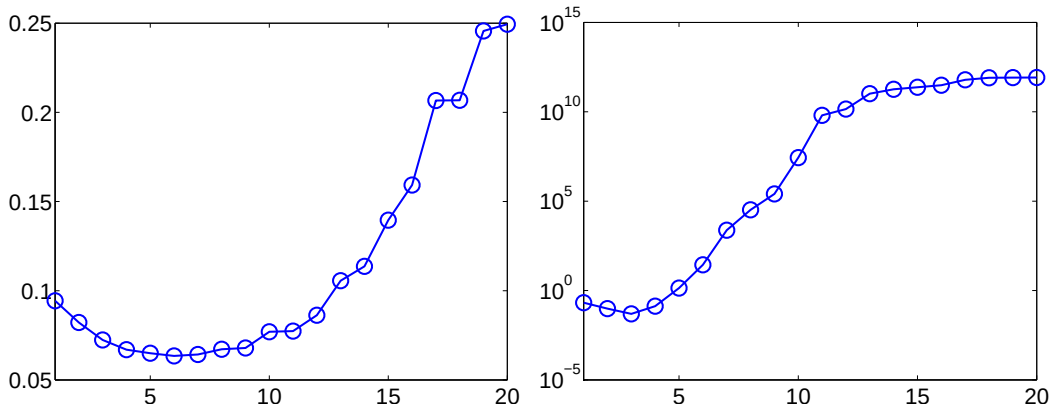

FIG. 2.1. $\delta_k = \|\mathbf{x}_{\mu_k} - \mathbf{x}_k\|$, *for* $k = 1, 2, \ldots, 20$, *for* deriv2-1 *(left-hand side) and* baart *(right-hand side) of size* 100×100 *with relative error* $\nu = 10^{-2}$ *in* $\mathbf{b}$.

Thus, the proposed choice of k is the first local minimum of a lower bound for the error norm $\|\mathbf{x}_k - \widehat{\mathbf{x}}\|$. To investigate whether in actual computations the inequality

$$\|\mathbf{x}_{\mu_k} - \mathbf{x}_k\| \leq 2\,\|\mathbf{x}_k - \widehat{\mathbf{x}}\| \tag{2.5}$$

typically holds, we solved 2400 linear discrete ill-posed problems of different kinds and sizes from Hansen's Regularization Tools [15] with relative errors $\nu = 10^{-3}$, 10^{-2}, and 10^{-1} in the vector $\mathbf{b}$; see (4.1) in Section 4. We found the inequality (2.5) to hold in 99.6% of all the examples. Further discussion on why the proposed choice of k may perform well is provided below.

Algorithm 1 Determination of regularization parameters

Input: Matrix $A \in \mathbb{R}^{m\times n}$ and data vector $\mathbf{b} \in \mathbb{R}^m$

Output: Tikhonov regularization parameter, TSVD truncation index, regularized solutions, and estimate of noise level

1: Compute SVD $A = U\Sigma V^T$
2: $\delta_0 = \infty$, $\mathbf{x}_0 = 0$, $k = 0$
3: **repeat**
4: $\quad k = k + 1$
5: $\quad$ Compute TSVD solution $\mathbf{x}_k = \mathbf{x}_{k-1} + \sigma_k^{-1}(\mathbf{u}_k^T\mathbf{b})\,\mathbf{v}_k$
6: $\quad$ Compute residual norm $\rho_k = \|\mathbf{b} - A\mathbf{x}_k\|$
7: $\quad$ Compute Tikhonov regularization parameter μ_k and solution $\mathbf{x}_{\mu_k} = \sum_{j=1}^{r} \frac{\sigma_j}{\sigma_j^2+\mu_k^2}\,(\mathbf{u}_j^T\mathbf{b})\,\mathbf{v}_j$ such that $\|\mathbf{b} - A\mathbf{x}_{\mu_k}\| = \rho_k$
8: $\quad$ Compute $\delta_k = \|\mathbf{x}_{\mu_k} - \mathbf{x}_k\|$
9: **until** $\delta_k > \delta_{k-1}$ **or** $k = n$
10: The TSVD truncation index is $k_{\min} = k - 1$, the noise level is approximately $\rho_{k_{\min}}/\|\mathbf{b}\|$, the Tikhonov regularization parameter is $\mu_{k_{\min}}$.

Algorithm 1 summarizes the computations required to determine the desired value of the regularization parameter k for TSVD, the associated ρ_k, and the corresponding value μ_k of the regularization parameter for Tikhonov regularization. We illustrate the performance of the algorithm with two experiments displayed in Figure 2.1. The

linear discrete ill-posed problem deriv2-1 from [15] (i.e., the MATLAB code deriv2 with the parameter example set to 1) is used to determine a matrix $A \in \mathbb{R}^{100\times 100}$ that is the discretization of an integral equation of the first kind; the integral equation is discussed in [9, p. 315]. The function deriv2 generates both A and the solution $\widehat{\mathbf{x}}$, from which we compute $\widehat{\mathbf{b}} = A\widehat{\mathbf{x}}$ and add 1% white Gaussian noise to obtain the noise contaminated vector $\mathbf{b}$, see (4.1), where we put $\nu = 10^{-2}$. Figure 2.1 (left-hand side) displays the differences $\|\mathbf{x}_{\mu_k} - \mathbf{x}_k\|$ for $k = 1, 2, \ldots, 20$. Algorithm 1 does not compute all of these values for efficiency reasons, but stops when the difference δ_k increases with k. Thus, the algorithm computes $\delta_1, \delta_2, \ldots, \delta_7$ and then suggests the noise level to be $\rho_6/\|\mathbf{b}\| = 1.03 \cdot 10^{-2}$. This is very close to the actual value, 10^{-2}, of the noise level.

Figure 2.1 (right-hand side plot) displays the result of a similar experiment for the baart test example with a matrix $A \in \mathbb{R}^{100\times 100}$ and the data vector $\mathbf{b} \in \mathbb{R}^{100}$ contaminated by 1% white Gaussian noise. This example is also from [15]; details about this problem are described in [1]. The suggested value $k = 3$ (after computation of the four values $\delta_1, \delta_2, \delta_3, \delta_4$) gives an accurate estimate $\rho_3/\|\mathbf{b}\| \approx 9.4 \cdot 10^{-3}$ of the norm of the relative error in $\mathbf{b}$. Note the logarithmic scale on the vertical axis. Additional examples can be found in Section 4.

We conclude this section with further comments on why the scheme implemented by Algorithm 1 is likely to yield a fairly accurate estimate of the noise level. Let

$$\gamma_j = \mathbf{u}_j^T \mathbf{b}, \qquad 1 \le j \le r.$$

In view of (1.2), we have

$$\widehat{\mathbf{x}} = \sum_{j=1}^{r} \sigma_j^{-1}\, \mathbf{v}_j\, (\mathbf{u}_j^T \widehat{\mathbf{b}}) = \sum_{j=1}^{r} \sigma_j^{-1}\, \mathbf{v}_j\, (\gamma_j - \mathbf{u}_j^T \mathbf{e}).$$

Writing $\mathbf{x}_k - \mathbf{x}_{\mu_k}$ as a linear combination of right singular vectors yields

$$\mathbf{x}_k - \mathbf{x}_{\mu_k} = \sum_{j=1}^{r} \eta_j^{(k)} \mathbf{v}_j,$$

and it follows from (1.5) and (1.7) that

$$\eta_j^{(k)} = \begin{cases} \gamma_j \left(\dfrac{1}{\sigma_j} - \dfrac{\sigma_j}{\sigma_j^2 + \mu_k^2} \right) = \dfrac{\mu_k^2}{\sigma_j^2} \cdot \dfrac{\gamma_j \sigma_j}{\sigma_j^2 + \mu_k^2}, & 1 \le j \le k, \\ -\dfrac{\gamma_j \sigma_j}{\sigma_j^2 + \mu_k^2}, & k+1 \le j \le r, \end{cases}$$

and, therefore,

$$\|\mathbf{x}_k - \mathbf{x}_{\mu_k}\|^2 = \sum_{j=1}^{r} \left(\eta_j^{(k)} \right)^2 = \sum_{j=1}^{k} \left(\frac{\mu_k}{\sigma_j} \right)^4 \cdot \left(\frac{\gamma_j \sigma_j}{\sigma_j^2 + \mu_k^2} \right)^2 + \sum_{j=k+1}^{r} \left(\frac{\gamma_j \sigma_j}{\sigma_j^2 + \mu_k^2} \right)^2. \tag{2.6}$$

We turn to the choice of μ_k. We have by (2.1) that

$$\rho_k^2 = \|\mathbf{b} - A\mathbf{x}_k\|^2 = \sum_{j=k+1}^{m} \gamma_j^2 \tag{2.7}$$

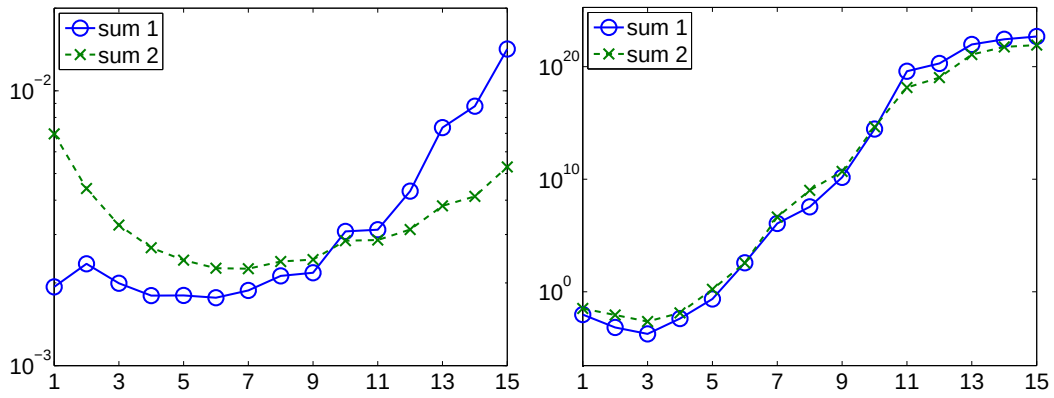

FIG. 2.2. *The sums in the right-hand side of (2.6) as a function of k. The continuous (blue) graph labeled "sum 1" depicts the first sum in the right-hand side of (2.6), the dashed (green) graph labeled "sum 2" the second sum. The graphs are for the problems* deriv2-1 *(left-hand side figure) and* baart *(right-hand side figure) with matrices of size 100×100 and with relative error $\nu = 10^{-2}$ in the data vector $\mathbf{b}$.*

and determine $\mu_k > 0$ so that

$$\rho_k^2 = \|\mathbf{b} - A\mathbf{x}_{\mu_k}\|^2 = \sum_{j=1}^{r} \left(1 + \frac{\sigma_j^2}{\mu_k^2}\right)^{-2} \gamma_j^2 + \sum_{j=r+1}^{m} \gamma_j^2. \tag{2.8}$$

Figure 2.2 shows the sums in the right-hand side of (2.6) as a function of k. The first sum in the right-hand side of (2.6) is shown by the continuous (blue) graph labeled "sum 1"; the second sum is depicted by the dashed (green) graph labeled "sum 2". The left-hand side figure is for the problem deriv2-1 and the right-hand side figure for the problem baart from [15]; see Section 4 for further comments on these problems. They define matrices of size 100×100. The data vector $\mathbf{b}$ has relative error $\nu = 10^{-2}$. Graphs for the square root of the left-hand side of (2.6) are shown in Figure 2.1. The sums in Figure 2.2 have unique local minima for $k \geq 2$. Figure 2.1 shows that the sum of the sums for each problem has a unique local minimum for $k \geq 1$.

Recall that Algorithm 1 determines the first local minimum $k_{\min}$ of $k \mapsto \|\mathbf{x}_k - \mathbf{x}_{\mu_k}\|$ for $k = 1, 2, 3, \ldots$. For many discrete ill-posed problems (1.1) this is also the global minimum, which typically is small; cf. the discussion in Section 1. We therefore may assume that $\mathbf{x}_{k_{\min}} \approx \mathbf{x}_{\mu_{k_{\min}}}$, which implies that expansions of $\mathbf{x}_{k_{\min}}$ and $\mathbf{x}_{\mu_{k_{\min}}}$ in terms of the left singular vectors $\mathbf{v}_j$ of A have about the same coefficients. In view of this, a comparison of the sums (2.7) and (2.8) suggests that

$$\frac{\mu_{k_{\min}}^2}{\sigma_j^2} \ll 1, \qquad 1 \leq j \leq k_{\min}, \tag{2.9}$$

$$\frac{\mu_{k_{\min}}^2}{\sigma_j^2} \gg 1, \qquad k_{\min} < j \leq r. \tag{2.10}$$

Assume that $\|\widehat{\mathbf{b}}\| = \mathcal{O}(1)$ and let $\varepsilon = \|\mathbf{e}\|$. Then $\|\widehat{\mathbf{x}}\| \approx \sigma_1^{-1}$. Since $\mathbf{e}$ represents white Gaussian noise, we obtain the "first-order approximations"

$$|\gamma_j| \approx \max\left\{\frac{\sigma_j}{\sigma_1}, \varepsilon\right\}, \qquad j = 1, 2, \ldots, r, \tag{2.11}$$

where we have ignored factors involving $\sqrt{m}$, because they are not important for our discussion. Let $k < r$ be determined by

$$\frac{\sigma_k}{\sigma_1} > \varepsilon \geq \frac{\sigma_{k+1}}{\sigma_1}. \tag{2.12}$$

We now consider an expression that is related to (2.6), but is simpler to analyze, in order to gain insight into properties of the index $k = k_{\min}$ determined by Algorithm 1. Substituting (2.11) into (2.6) with $\mu_k = \mu_{k_{\min}}$ yields

$$\|\mathbf{x}_k - \mathbf{x}_{\mu_{k_{\min}}}\|^2 \approx \sigma_1^{-2} \left(\sum_{j=1}^{k} \left(\frac{\mu_{k_{\min}}}{\sigma_j} \right)^4 \cdot \left(1 + \frac{\mu_{k_{\min}}^2}{\sigma_j^2} \right)^{-2} + \varepsilon^2 \sum_{j=k+1}^{r} \left(\frac{\sigma_j}{\sigma_j^2 + \mu_{k_{\min}}^2} \right)^2 \right). \tag{2.13}$$

The right-hand side provides some insight into properties of the index $k = k_{\min}$ determined by Algorithm 1. Consider the first sum on the right-hand side of (2.13). Its terms are determined by the function

$$h_1(t) = \frac{t^4}{(1+t^2)^2}, \qquad t \geq 0,$$

which is monotonically increasing with $h_1(0) = 0$ and $\lim_{t\to\infty} h_1(t) = 1$. It follows from (2.9) and (2.10) that the sum is much smaller when $k \leq k_{\min}$ than when $k > k_{\min}$.

We turn to the second sum on the right-hand side of (2.13). Its terms are determined by the function

$$h_2(t) = \frac{t}{t^2 + \mu_{k_{\min}}^2}, \qquad t \geq 0,$$

with $\mu_{k_{\min}} > 0$. This function is increasing for $0 \leq t \leq \mu_{k_{\min}}$ and decreasing for $t \geq \mu_{k_{\min}}$. We have $h_2(0) = \lim_{t\to\infty} h_2(t) = 0$. The maximum $h_2(\mu_{k_{\min}}) = (2\mu_{k_{\min}})^{-1}$ is large when $\mu_{k_{\min}}$ is small. The inequality (2.10) indicates that $h_2(\sigma_j)$ is not very large for $j > k_{\min}$; however, $h_2(\sigma_j)$ may be large when $j \leq k_{\min}$ and $\mu_{k_{\min}} > 0$ is small. It follows that the second sum on the right-hand side of (2.13) is not very large when $k \geq k_{\min}$, but may be large when $\mu_{k_{\min}}$ is small and $k < k_{\min}$.

The above discussion indicates that to a "first-order approximation" both sums in the right-hand side of (2.13) are fairly small when $k = k_{\min}$, i.e., when (2.12) holds for $k = k_{\min}$, where $k_{\min}$ is determined by Algorithm 1. Thus, $k_{\min}$ is the smallest index k such that (2.6) has a local minimum. The left-hand sides of (2.6) and (2.13) differ, but they are close when k is close to $k_{\min}$. Our analysis suggests that to "first-order approximation" $\sigma_{k_{\min}}/\sigma_1 \approx \varepsilon$; i.e., $\sigma_{k_{\min}}/\sigma_1$ is roughly of the order of the relative error in the data $\mathbf{b}$. Table 2.1 illustrate that this is indeed the case for several linear discrete ill-posed problems from [15]. The problems have matrices of size 100×100 and relative errors 10^{-3}, 10^{-2}, and 10^{-1} in $\mathbf{b}$. The table also depicts the best possible truncation index k_{best} for TSVD. It satisfies

$$\|\mathbf{x}_{k_{\text{best}}} - \widehat{\mathbf{x}}\| = \min_{j\geq 1} \|\mathbf{x}_j - \widehat{\mathbf{x}}\|.$$

In case of non-uniqueness, the smallest such index is shown. Let k_σ denote the index k that satisfies (2.12). Table 2.1 shows that for many problems and several noise levels,

TABLE 2.1

Three truncation indices for TSVD for several linear discrete ill-posed problems and three noise levels.

	10^{-3}			10^{-2}			10^{-1}		
Method	k_{best}	$k_{\min}$	k_σ	k_{best}	$k_{\min}$	k_σ	k_{best}	$k_{\min}$	k_σ
baart	4	4	4	3	3	3	3	2	2
foxgood	3	3	4	2	2	2	2	2	2
hilbert	6	5	5	4	5	4	4	3	3
ilaplace(3)	1	1	1	1	1	1	1	1	1
lotkin	5	4	5	4	3	3	3	2	2
shaw	7	7	8	7	4	6	5	4	4

k_σ is an accurate approximation of k_{best}. In fact, for a few of the problems shown, k_σ is closer to k_{best} than to $k_{\min}$. We remark that k_σ is introduced to illustrate the performance of Algorithm 1, but cannot be used to determine a truncation index when the noise level ε is not known, because the computation of k_σ requires knowledge of ε.

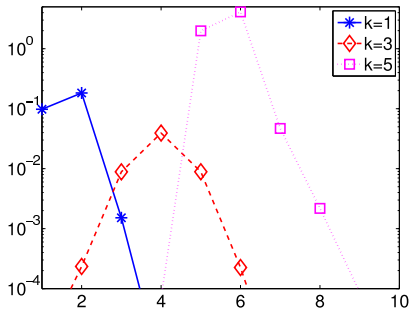

FIG. 2.3. $|\eta_j^{(k)}|$*-factors for* $j = 1, 2, \ldots, 10$ *for the solutions for which* $k = 1$ *(asterisks),* $k = 3$ *(diamonds), and* $k = 5$ *(squares) for* baart *of size* 100×100 *with relative error* $\nu = 10^{-2}$ *in* $\mathbf{b}$.

We now discuss another aspect of how Algorithm 1 determines the index $k = k_{\min}$. Consider the example on the right-hand side of Figure 2.1. The figure shows the index $k = 1$ to be too small, the index $k = 5$ to be too large, and the index $k = 3$ to be optimal. When k is too small, the residual $\mathbf{b} - A\mathbf{x}_k$ is too large, which implies that the regularization parameter $\mu = \mu_k$ is large. This indicates that the first few coefficients $\eta_j^{(k)}$ typically are of fairly large magnitude, since for the first few indices σ_j^{-1} generally will be clearly larger than $\frac{\sigma_j}{\sigma_j^2+\mu_k^2}$. This is illustrated by the asterisks in Figure 2.3. Conversely, if k is too large, then the residual error $\mathbf{b} - A\mathbf{x}_k$ is too small. This situation is depicted by the squares in Figure 2.3. For the optimal value of k, the maximal value of $|\eta_j^{(k)}|$ tends to be smaller than when k is either too large or too small; see the diamonds in Figure 2.3.

As long as $j \leq k$, we have in view of (2.11) that, roughly,

$$|\eta_j^{(k)}| = \frac{|\gamma_j|\,\mu_k^2}{\sigma_j\,(\sigma_j^2 + \mu_k^2)} \approx \frac{\mu_k^2}{\sigma_j^2 + \mu_k^2}.$$

Therefore, we expect the $|\eta_j^{(k)}|$ to be increasing with j for $j \leq k$. If instead $j > k$ and

$\sigma_j > \varepsilon$, then we have approximately

$$|\eta_j^{(k)}| = \frac{|\gamma_j|\,\sigma_j}{\sigma_j^2 + \mu_k^2} \approx \frac{\sigma_j^2}{\sigma_j^2 + \mu_k^2},$$

which is a decreasing function of j. This discussion suggests that $|\eta_j^{(k)}|$ may be maximal for some $j \approx k$. Indeed, $j \to |\eta_j^{(k)}|$ is maximal for $j = k+1$ for all three graphs of Figure 2.3.

3. Large-scale problems. Algorithm 1 computes an SVD of the matrix A. This is not practical for large-scale problems. Instead, we reduce large-scale least-squares problems (1.1) to small size by carrying out a few steps of Golub–Kahan bidiagonalization. Application of k steps of Golub–Kahan bidiagonalization to A with initial vector $\mathbf{b}$ yields the decompositions

$$A\widetilde{V}_k = \widetilde{U}_{k+1}\overline{C}_k, \qquad A^T\widetilde{U}_k = \widetilde{V}_k C_k, \tag{3.1}$$

where $\widetilde{U}_{k+1} = [\widetilde{\mathbf{u}}_1, \widetilde{\mathbf{u}}_2, \ldots, \widetilde{\mathbf{u}}_{k+1}] \in \mathbb{R}^{m\times(k+1)}$ and $\widetilde{V}_k = [\widetilde{\mathbf{v}}_1, \widetilde{\mathbf{v}}_2, \ldots, \widetilde{\mathbf{v}}_k] \in \mathbb{R}^{n\times k}$ have orthonormal columns with $\widetilde{\mathbf{u}}_1 = \mathbf{b}/\|\mathbf{b}\|$, $\widetilde{U}_k \in \mathbb{R}^{m\times k}$ consists of the first k columns of $\widetilde{U}_{k+1}$,

$$\overline{C}_k = \begin{bmatrix} \alpha_1 & & & \\ \beta_2 & \alpha_2 & & \\ & \ddots & \ddots & \\ & & \beta_k & \alpha_k \\ & & & \beta_{k+1} \end{bmatrix} \in \mathbb{R}^{(k+1)\times k} \tag{3.2}$$

is lower bidiagonal, and C_k is the leading $k \times k$ submatrix of $\overline{C}_k$. Moreover, the columns of $\widetilde{V}_k$ span the Krylov subspace

$$\mathcal{K}_k(A^TA, A^T\mathbf{b}) = \operatorname{span}\{A^T\mathbf{b}, (A^TA)A^T\mathbf{b}, \ldots, (A^TA)^{k-1}A^T\mathbf{b}\}; \tag{3.3}$$

see, e.g., [11, 21] for details. We will assume that the nontrivial entries of the bidiagonal matrices (3.2) are positive for all k. This is the generic situation.

LSQR [21] is an iterative method based on the decomposition (3.1). Let the initial iterate be $\mathbf{x}_0 = \mathbf{0}$. The kth iterate $\mathbf{x}_k$ determined by the LSQR satisfies

$$\|A\mathbf{x}_k - \mathbf{b}\| = \min_{\mathbf{x}\in\mathcal{K}_k(A^TA, A^T\mathbf{b})} \|A\mathbf{x} - \mathbf{b}\|, \qquad \mathbf{x}_k \in \mathcal{K}_k(A^TA, A^T\mathbf{b}). \tag{3.4}$$

This shows that LSQR is a so-called minimal residual method: the iterate $\mathbf{x}_k$ minimizes the residual error over the Krylov subspace (3.3). Substituting $\mathbf{x} = \widetilde{V}_k\mathbf{y}$ into the right-hand side of (3.4) and using the decompositions (3.1) shows that the kth iterate can be expressed as $\mathbf{x}_k = \widetilde{V}_k\,\mathbf{y}_k$, where $\mathbf{y}_k$ is the solution of the reduced problem

$$\min_{\mathbf{y}\in\mathbb{R}^k} \left\|\overline{C}_k\mathbf{y} - \beta_1\mathbf{e}_1\right\|, \qquad \beta_1 = \|\mathbf{b}\|.$$

We have

$$\rho_k = \|\overline{C}_k\mathbf{y}_k - \beta_1\mathbf{e}_1\| = \|A\mathbf{x}_k - \mathbf{b}\|. \tag{3.5}$$

Assume that we have carried out $\ell \geq k$ Golub–Kahan bidiagonalization steps and have available decompositions (3.1) with k replaced by ℓ. Substituting these decompositions into (1.4) yields the reduced Tikhonov minimization problem

$$\min_{\mathbf{y}\in\mathbb{R}^\ell} \left\| \begin{bmatrix} \overline{C}_\ell \\ \mu I_\ell \end{bmatrix} \mathbf{y} - \beta_1 \mathbf{e}_1 \right\|. \tag{3.6}$$

It has a unique solution $\mathbf{y}_{\mu,\ell}$ for any $\mu > 0$. An approximation of the solution (1.5) of (1.4) is furnished by $\mathbf{x}_{\mu,\ell} = \widetilde{V}_\ell \mathbf{y}_{\mu,\ell}$. This vector satisfies a Galerkin equation determined by the normal equations associated with (1.4) and the space range($\widetilde{V}_\ell$); see, e.g., [6] for details. The following result is analogous to Proposition 2.1.

PROPOSITION 3.1. *Introduce the function*

$$\varphi_\ell(\mu) = \|\mathbf{b} - A\mathbf{x}_{\mu,\ell}\|^2, \qquad \mu > 0,$$

where $\mathbf{x}_{\mu,\ell} = \widetilde{V}_\ell \mathbf{y}_{\mu,\ell}$ *and* $\mathbf{y}_{\mu,\ell}$ *solves (3.6). Then* $\varphi_\ell(\mu)$ *is a strictly increasing function of* μ*. Moreover, the equation*

$$\varphi_\ell(\mu) = \gamma \tag{3.7}$$

has a unique solution μ*, such that* $0 < \mu < \infty$*, for any* $0 < \gamma < \|\mathbf{b}\|^2$*.*

Proof. Similarly as in the proof of Proposition 2.1, we can write $\varphi_\ell(\mu)$ in the form

$$\varphi_\ell(\mu) = \|\mathbf{b}\|^2 \mathbf{e}_1^T (\mu^{-2} \overline{C}_\ell \overline{C}_\ell^T + I_\ell)^{-2} \mathbf{e}_1.$$

This representation shows that φ_ℓ is a strictly increasing function of $\mu > 0$. The stated properties of the equation (3.7) follow from the observation that the matrix $\overline{C}_\ell$ is of full rank. □

We would like to determine a solution $\mathbf{y}_{\mu,\ell}$ of the reduced problem (3.6) that achieves the same residual error norm as the LSQR solution $\mathbf{x}_k$, i.e., we determine the regularization parameter $\mu = \mu_k$ in (3.6) so that

$$\|\overline{C}_\ell \mathbf{y}_{\mu_k,\ell} - \beta_1 \mathbf{e}_1\| = \rho_k, \tag{3.8}$$

where ρ_k is defined by (3.5). Since the LSQR solution $\mathbf{x}_k$ minimizes the norm of the residual error over the Krylov subspace (3.3), we have to choose $\ell > k$ in order to be able to satisfy (3.8) for some $\mu = \mu_k > 0$. We increase ℓ until two consecutive solutions $\mathbf{y}_{\mu,\ell-1}$ and $\mathbf{y}_{\mu,\ell}$ of reduced Tikhonov minimization problems are sufficiently close. The approximate solution of (1.4) corresponding to $\mathbf{y}_{\mu_k,\ell}$ is given by $\mathbf{x}_{\mu_k,\ell} = \widetilde{V}_\ell \mathbf{y}_{\mu_k,\ell}$.

The computations are described by Algorithms 2 and 3. The former algorithm initializes the method and the latter contains the main loop. The index k in the loop corresponds to the LSQR solution, while the index ℓ counts the number of Golub–Kahan bidiagonalization steps required for computing the Tikhonov solution. We increase k (and also ℓ) until a local minimum of the differences

$$\delta_k = \|\mathbf{x}_k - \mathbf{x}_{\mu_k,\ell}\| = \|\mathbf{y}_k - \mathbf{y}_{\mu_k,\ell}\|$$

has been found. To avoid the effect of small fluctuations near the minimum that occur in some examples due to rounding errors, we terminate the iterations when 4 consecutive increasing values of δ_k are detected. The regularization parameter μ is

Algorithm 2 Large-scale computation, part 1: initialization

Input: Matrix A and data vector $\mathbf{b}$, $\tau(=10^{-4})$, $N_{\max}(=50)$
Output: Regularization parameter p, LSQR solution $\mathbf{x}$, estimate of noise level s

1: $\beta_1 = \|\mathbf{b}\|$
2: Perform the first bidiagonalization step: compute $\mathbf{v}_1$, $\mathbf{u}_1$, $\mathbf{u}_2$, α_1, β_2, $\overline{C}_1$
3: Solve $\min_{\mathbf{y}\in\mathbb{R}^2} \|\overline{C}_1\mathbf{y} - \beta_1\mathbf{e}_1\|$ for $\mathbf{y}_1$ {Compute the first LSQR solution}
4: $\rho_1 = \|\overline{C}_1\mathbf{y}_1 - \beta_1\mathbf{e}_1\|$ {Compute the first residual}
5: $\ell = 1$, $\mu = 1$
6: Solve $\min\limits_{\mathbf{y}\in\mathbb{R}^2} \left\| \begin{bmatrix} \overline{C}_1 \\ \mu \end{bmatrix} \mathbf{y} - \beta_1\mathbf{e}_1 \right\|$ for $\mathbf{y}_\mu$
7: **repeat**
8: $\quad \ell = \ell + 1$
9: $\quad \mathbf{y}_{\text{old}} = \begin{bmatrix} \mathbf{y}_\mu \\ 0 \end{bmatrix}$
10: $\quad$ Perform the ℓth bidiagonalization step: compute $\mathbf{v}_\ell$, $\mathbf{u}_{\ell+1}$, α_ℓ, $\beta_{\ell+1}$, $\overline{C}_\ell$
11: $\quad$ Solve $\min\limits_{\mathbf{y}\in\mathbb{R}^\ell} \left\| \begin{bmatrix} \overline{C}_\ell \\ \mu I_\ell \end{bmatrix} \mathbf{y} - \beta_1\mathbf{e}_1 \right\|$ for $\mathbf{y}_\mu$
12: **until** $\|\mathbf{y}_{\text{old}} - \mathbf{y}_\mu\| < \tau\,\|\mathbf{y}_\mu\|$ **or** $\ell > N_{\max}$
13: Find μ and $\mathbf{y}_\mu$ such that $\left\| \begin{bmatrix} \overline{C}_\ell \\ \mu I_\ell \end{bmatrix} \mathbf{y}_\mu - \beta_1\mathbf{e}_1 \right\| = \rho_1$
14: $\delta_1 = \|\widetilde{\mathbf{y}}_1 - \mathbf{y}_\mu\|$ {$\widetilde{\mathbf{y}}_1$ is zero-padded to the size of $\mathbf{y}_\mu$}

determined by Newton's method in lines 13 and 34. Note that the determination of μ_k and δ_k is performed in low-dimensional setting.

We illustrate the performance of Algorithms 2–3 in Section 4. Other approaches to reduce large problems to small ones for which the regularization parameters can be determined in an analogous manner as described can be devised. Experiments with small to medium-sized discrete ill-posed problems with Algorithm 1 indicate that typically only few singular triplets are required to determine an approximate solution. This suggests alternative approaches to Algorithms 2–3 for large-scale problems. For instance, it may be attractive to compute a partial singular value decomposition involving the largest singular values of A. Such a decomposition can be determined by repeatedly computing a partial Golub–Kahan bidiagonalization of A as described in [2].

4. Numerical examples. This and the following sections present several numerical experiments that show the performance of the proposed methods to determine suitable regularization parameters. In this section, we first compute solutions of the linear discrete ill-posed problems listed in Table 4.1. These problems are from Hansen's Regularization Tools [15] and from the gallery function of MATLAB. Each problem from [15] comes with a solution $\widehat{\mathbf{x}}$; for the gallery examples, we use the solution of the problem shaw from [15]. We determine the noise free data vector as $\widehat{\mathbf{b}} = A\widehat{\mathbf{x}}$; the associated perturbed data vector $\mathbf{b}$ is obtained by adding a "noise vector" $\mathbf{e}$ to $\widehat{\mathbf{b}}$; cf. (1.2). Specifically, let the vector $\mathbf{w}$ have normally distributed entries with mean zero and variance one, and compute

$$\mathbf{b} = \widehat{\mathbf{b}} + \mathbf{w}\,\|\widehat{\mathbf{b}}\|\frac{\nu}{\sqrt{n}}. \tag{4.1}$$

Algorithm 3 Large-scale computation, part 2: main loop

$k = 1$
repeat
  $k = k + 1$
  if $k > \ell$ **then**
    $\ell = \ell + 1$
    Perform the ℓth bidiagonalization step: compute $\mathbf{v}_\ell$, $\mathbf{u}_{\ell+1}$, α_ℓ, $\beta_{\ell+1}$, $\overline{C}_\ell$
  end if
  Compute QR factorization $\overline{C}_k = QR$
  $\mathbf{c} = \beta_1 Q^T \mathbf{e}_1$
  Solve $R_{1:k,1:k}\mathbf{y}_k = \mathbf{c}_{1:k}$ {Compute the kth LSQR solution}
  $\rho_k = |c_{k+1}|$ {Compute the kth residual}
  Solve $\min_{\mathbf{y}\in\mathbb{R}^{\ell-1}} \left\| \begin{bmatrix} \overline{C}_{\ell-1} \\ \mu I_{\ell-1} \end{bmatrix} \mathbf{y} - \beta_1 \mathbf{e}_1 \right\|$ for $\mathbf{y}_\mu$, $\mathbf{y}_{\text{old}} = \begin{bmatrix} \mathbf{y}_\mu \\ 0 \end{bmatrix}$
  Solve $\min_{\mathbf{y}\in\mathbb{R}^{\ell}} \left\| \begin{bmatrix} \overline{C}_{\ell} \\ \mu I_{\ell} \end{bmatrix} \mathbf{y} - \beta_1 \mathbf{e}_1 \right\|$ for $\mathbf{y}_\mu$
  while $\|\mathbf{y}_{\text{old}} - \mathbf{y}_\mu\| \geq \tau \|\mathbf{y}_\mu\|$ **and** $\ell < k + N_{\max}$ **do**
    $\ell = \ell + 1$
    $\mathbf{y}_{\text{old}} = \begin{bmatrix} \mathbf{y}_\mu \\ 0 \end{bmatrix}$
    Perform the ℓth bidiagonalization step: compute $\mathbf{v}_\ell$, $\mathbf{u}_{\ell+1}$, α_ℓ, $\beta_{\ell+1}$, $\overline{C}_\ell$
    Solve $\min_{\mathbf{y}\in\mathbb{R}^{\ell}} \left\| \begin{bmatrix} \overline{C}_{\ell} \\ \mu I_{\ell} \end{bmatrix} \mathbf{y} - \beta_1 \mathbf{e}_1 \right\|$ for $\mathbf{y}_\mu$
  end while
  Find μ and $\mathbf{y}_\mu$ such that $\left\| \begin{bmatrix} \overline{C}_{\ell} \\ \mu I_{\ell} \end{bmatrix} \mathbf{y}_\mu - \beta_1 \mathbf{e}_1 \right\| = \rho_k$
  $\delta_k = \|\widetilde{\mathbf{y}}_k - \mathbf{y}_\mu\|$ {$\widetilde{\mathbf{y}}_k$ is zero-padded to the size of $\mathbf{y}_\mu$}
  {Iteration stops when 4 increasing δ_k are found}
  if $\delta_k > \delta_{k-1}$ **then** $m = m + 1$ **else** $m = 0$ **endif**
until $m \geq 4$ **or** $k > N_{\max}$
$p = \arg\min \delta_k$, $s = \rho_p / \|\mathbf{b}\|$
$\mathbf{x} = V_{:,1:p}\, \mathbf{y}_p$

Then

$$\frac{\|\mathbf{b} - \widehat{\mathbf{b}}\|}{\|\widehat{\mathbf{b}}\|} \approx \nu.$$

In the computed examples we use the noise levels $\nu = 10^{-3}$, $\nu = 10^{-2}$, and $\nu = 10^{-1}$, which are compatible with real world applications. Algorithm 1 and Algorithms 2–3 have been implemented in MATLAB; code is available from the authors upon request.

Experiment 4.1. The aim of this example is to illustrate the performance of Algorithm 1. We consider the linear discrete ill-posed problems listed in Table 4.1. Each problem is discretized to give two examples, one with a matrix of size 40×40 and one with a matrix of size 100×100. We generate 10 realizations of the noise vector $\mathbf{e}$ for each of the three noise levels and each matrix. This gives 600 linear discrete

ill-posed problems. Table 4.1 displays noise level ratios

$$\frac{\text{estimated noise level}}{\text{true noise level}} = \frac{\rho_k}{\nu\,\|\mathbf{b}\|}, \tag{4.2}$$

where k is the index that yields the first local minimum of (2.4). Each reported ratio is the average over 20 tests; 10 for each matrix size. In an ideal situation, this ratio is one. The ratios reported in Table 4.1 are between 0.735 and 1.344, but the vast majority of them are very close to one. Table 4.1 shows that ρ_k determined by Algorithm 1 is a quite reliable noise level estimator. The computations required to evaluate this estimator are negligible in comparison with the evaluation of the SVD of the matrix.

TABLE 4.1

Noise-level ratios obtained by estimating the TSVD truncation parameter by the method of the present paper for 10 *examples and noise levels* 10^{-3}, 10^{-2}, 10^{-1}.

Problem	10^{-3}	10^{-2}	10^{-1}
baart	1.008	1.005	1.005
deriv2(2)	1.117	1.190	1.069
foxgood	1.010	1.015	1.006
gravity	0.964	0.996	0.991
heat(1)	0.735	0.981	1.344
hilbert	0.995	1.006	1.006
ilaplace(3)	0.994	1.219	0.997
lotkin	1.041	1.007	1.002
phillips	0.972	0.966	0.979
shaw	0.973	1.039	0.999

Any parameter estimation criterion may be considered a method for estimating the noise level in the available data, with the norm of residual error furnishing an estimate of the norm of the noise. We therefore provide tables analogous to Table 4.1 for popular methods for determining a suitable value of the regularization parameter. Table 4.2 reports results obtained by determining the TSVD truncation parameter by the L-corner algorithm discussed in [17] (columns 2 to 4) and by the quasi-optimality criterion (columns 5 to 7). The latter criterion is discussed in, e.g., [10, 14, 19, 24]. In both cases the regularization parameter was computed with an appropriate MATLAB function from [15]. Having determined the regularization parameter k and the associated approximate solution $\mathbf{x}_k$ of (1.1), we compute the norm of the residual error $\|A\mathbf{x}_k - \mathbf{b}\|$ and use it to evaluate the noise level ratio (4.2). Table 4.2 displays these ratios. Table 4.3 considers Tikhonov regularization and determines the regularization parameter μ by the L-curve method, using the MATLAB function l_curve from [15]. Let $\mathbf{x}_\mu$ denote the associated approximate solution of (1.1). Then $\|A\mathbf{x}_\mu - \mathbf{b}\|$ provides an estimate for the norm of the noise. Columns 2–4 tabulate the noise level ratios (4.2) determined in this manner. The three right-most columns of Table 4.2 are obtained by using the quasi-optimality criterion.

These tables indicate that the method for determining regularization parameters of this paper is more a trustworthy noise level estimator than the other methods for determining the regularization parameter in our comparison. This is confirmed by Table 4.4, which displays the standard deviation of the ratios obtained by each

method with respect to the expected value one. In this and in the following tables the method of this paper is referred to as COSE (comparison of solutions estimator).

TABLE 4.2
Noise-level ratios for 10 *examples and noise levels* 10^{-3}, 10^{-2}, 10^{-1}, *determined by estimating the TSVD truncation parameter by the L-corner method (columns 2–4) and the quasi-optimality criterion (columns 5–7).*

Problem	10^{-3}	10^{-2}	10^{-1}	10^{-3}	10^{-2}	10^{-1}
baart	0.994	1.005	1.006	1.027	1.267	1.138
deriv2(2)	0.737	0.944	1.005	0.740	0.903	0.976
foxgood	0.998	1.015	1.006	1.022	1.008	1.054
gravity	0.918	0.952	0.986	0.975	0.993	0.993
heat(1)	0.432	0.549	0.925	0.717	3.454	0.960
hilbert	0.990	0.998	1.030	0.999	1.018	1.475
ilaplace(3)	0.923	0.937	0.935	4.165	1.330	0.996
lotkin	1.000	1.005	1.002	1.304	1.019	1.026
phillips	0.779	0.926	0.971	362.4	36.27	3.601
shaw	0.960	0.977	0.999	1.370	0.995	1.329

TABLE 4.3
Noise-level ratios for 10 *examples and noise levels* 10^{-3}, 10^{-2}, 10^{-1}, *determined by estimating the optimal Tikhonov regularization parameter by the L-curve method (columns 2–4) and the quasi-optimality criterion (columns 5–7).*

Problem	10^{-3}	10^{-2}	10^{-1}	10^{-3}	10^{-2}	10^{-1}
baart	0.995	1.006	1.007	1.032	1.043	1.009
deriv2(2)	0.729	0.920	1.005	1.195	1.214	1.147
foxgood	0.998	1.008	1.010	1.116	1.026	1.018
gravity	0.911	0.949	0.982	1.037	1.057	1.013
heat(1)	0.425	0.540	0.419	0.161	0.161	0.161
hilbert	0.989	0.997	1.010	1.003	1.020	1.261
ilaplace(3)	0.978	0.993	0.995	1.007	1.373	1.002
lotkin	0.999	1.004	1.006	1.117	1.012	1.038
phillips	0.774	0.907	0.962	1.442	1.053	1.006
shaw	0.956	0.977	1.000	1.177	1.119	1.009

TABLE 4.4
Standard deviations of noise level estimates reported in Tables 4.1, 4.2, and 4.3.

COSE	TSVD/L-corner	TSVD/Quasiopt	Tikh/L-curve	Tikh/Quasiopt
0.099	0.136	66.009	0.168	0.305

An alternative to letting k be the first local minimum of (2.4) when increasing k from $k = 1$ is to consider the first local minimum of the weighted difference

$$\widetilde{\delta}_k = \|\mathbf{x}_{\mu_k} - \mathbf{x}_k\| \,/\, \|\mathbf{x}_k\|. \tag{4.3}$$

For some problems $\widetilde{\delta}_k$ has a deeper first local minimum than δ_k, but also may be a more oscillatory function of k. Nevertheless, on average we obtain slightly better noise

level estimates using (4.3) than (2.4). This is illustrated by Table 4.5. The analog of the entry COSE of Table 4.4 is 0.080.

TABLE 4.5

Noise-level ratios obtained by estimating the TSVD truncation parameter. The table is analogous to Table 4.1. The truncation parameter k is the first local minimum of (4.3), while it is the first local minimum of (2.4) in Table 4.1.

Problem	10^{-3}	10^{-2}	10^{-1}
baart	1.008	1.005	0.994
deriv2(2)	1.109	1.154	1.053
foxgood	1.010	1.015	1.006
gravity	0.964	0.996	0.991
heat(1)	0.735	0.928	1.225
hilbert	0.995	1.006	0.996
ilaplace(3)	0.994	1.115	0.974
lotkin	1.041	1.007	1.001
phillips	0.972	0.966	0.979
shaw	0.973	1.039	0.999

Table 4.6 compares COSE with several methods for determining the truncation index in TSVD. The latter methods are described in detail in [24]. They include the methods L-corner [17], Res L-curve [26], Cond L-curve [7], Regińska [23], and ResReg [24], all of which are related to the L-curve criterion [13]. Other methods in our comparison are the quasi-optimality criterion, generalized cross validation (GCV) [14], extrapolation [5], and the discrepancy principle (1.8) with $\tau = 1.3$. While the focus of this paper is on heuristic parameter choice rules, it is interesting to compare these rules with the discrepancy principle. Tables 4.6 and 4.7 show that the COSE method on average performs better than the discrepancy principle. Thus, it may be worthwhile to determine the regularization parameter with COSE, also when an estimate for the error in the data vector $\mathbf{b}$ is known.

Let k_{best} denote the truncation index that yields the smallest error, i.e.,

$$\|\mathbf{x}_{k_{\text{best}}} - \widehat{\mathbf{x}}\| = \min_j \|\mathbf{x}_j - \widehat{\mathbf{x}}\|.$$

The first entries in the second to fifth columns of Table 4.6 show the percentage of experiments that produced an approximate solution $\mathbf{x}_k$ with

$$\|\mathbf{x}_k - \widehat{\mathbf{x}}\| > 2\,\|\mathbf{x}_{k_{\text{best}}} - \widehat{\mathbf{x}}\|; \tag{4.4}$$

the second entries (in parentheses) display the percentage of experiments that produced an approximate solution $\mathbf{x}_k$ with

$$\|\mathbf{x}_k - \widehat{\mathbf{x}}\| > 5\,\|\mathbf{x}_{k_{\text{best}}} - \widehat{\mathbf{x}}\|. \tag{4.5}$$

The results concerning the above set of 600 square linear systems are reported in the second column of Table 4.6. The last three columns are obtained when solving the same number of overdetermined least-squares problems with a matrix $A \in \mathbb{R}^{2n\times n}$ for $n = 40$ and $n = 100$. These problems are either consistent ($\xi = 0$) or inconsistent ($\xi = 1$ or 10). To make a least-squares problem inconsistent, we add to the exact data

TABLE 4.6
Parameter choice rules for TSVD; factors 2 *and* 5.

	Square	Overdetermined systems		
Method	systems	$\xi = 0$	$\xi = 1$	$\xi = 10$
COSE	6% (0%)	7% (1%)	7% (1%)	8% (1%)
L-corner	24% (12%)	23% (10%)	36% (10%)	63% (40%)
Res L-curve	26% (6%)	35% (1%)	61% (32%)	71% (47%)
Cond L-curve	40% (5%)	50% (11%)	69% (46%)	67% (49%)
Regińska	25% (10%)	30% (11%)	58% (35%)	76% (51%)
ResReg	24% (8%)	30% (11%)	58% (35%)	76% (51%)
Quasiopt	31% (14%)	31% (14%)	31% (14%)	28% (12%)
GCV	29% (22%)	17% (10%)	40% (18%)	67% (36%)
Extrapolation	62% (14%)	76% (34%)	83% (62%)	76% (52%)
Discrepancy	17% (1%)	38% (4%)	62% (37%)	71% (51%)

TABLE 4.7
Parameter choice rules for TSVD; factors 10 *and* 100.

	Square	Overdetermined systems		
Method	systems	$\xi = 0$	$\xi = 1$	$\xi = 10$
COSE	0% (0%)	0% (0%)	0% (0%)	0% (0%)
L-corner	7% (0%)	4% (0%)	5% (1%)	27% (11%)
Res L-curve	3% (1%)	1% (0%)	18% (5%)	34% (18%)
Cond L-curve	3% (2%)	7% (5%)	32% (18%)	38% (23%)
Regińska	4% (0%)	2% (0%)	13% (0%)	24% (1%)
ResReg	3% (0%)	2% (0%)	13% (0%)	24% (1%)
Quasiopt	9% (1%)	6% (1%)	6% (1%)	5% (1%)
GCV	20% (16%)	8% (4%)	8% (0%)	11% (0%)
Extrapolation	5% (1%)	9% (0%)	35% (0%)	25% (1%)
Discrepancy	0% (0%)	0% (0%)	31% (31%)	41% (40%)

vector $\widehat{\mathbf{b}}$ a multiple ξ of a unit vector $\mathbf{q} \in \mathbb{R}^{2n}$ that is orthogonal to the range of A, so that

$$\min_{\mathbf{x}\in\mathbb{R}^n} \|A\mathbf{x} - (\widehat{\mathbf{b}} + \xi\mathbf{q})\| = \xi.$$

Tables 4.6 and 4.7 (where the factors 2 and 5 in (4.4) and (4.5) are replaced by 10 and 100, respectively) show that COSE produces very good results, failing to determine the best possible solution in only a small number of tests. In particular, the method is effective also for inconsistent problems. To the best of our knowledge, no other available method gives comparable results for such problems.

We note that the measurements ascribed to the discrepancy principle are obtained by selecting the smallest integer k such that

$$\|A\mathbf{x}_k - \mathbf{b}\|^2 \le (1.3\,\nu\,\|\mathbf{b}\|)^2 + \xi^2.$$

We also remark that the results in Table 4.6 do not imply that we are obtaining very accurate approximations of the desired solution $\widehat{\mathbf{x}}$ for all the least-squares problems,

but rather that Algorithm 1 computes optimal or nearly optimal approximations of $\hat{\mathbf{x}}$, i.e., the algorithm determines approximations of $\widehat{\mathbf{x}}$ that are the best possible or close to the best possible that can be computed with the TSVD method.

TABLE 4.8
Average number of LSQR iterations required by our method and average number of Golub–Kahan bidiagonalization steps performed when solving 10 *examples for different noise levels (*10^{-3}, 10^{-2}, 10^{-1}*).*

Problem	10^{-3}	10^{-2}	10^{-1}
baart	4/9	3/8	3/8
deriv2(2)	12/26	7/18	3/14
foxgood	3/9	3/9	2/9
gravity	9/17	7/16	5/14
heat(1)	24/39	16/30	10/23
hilbert	7/13	6/13	5/12
ilaplace(3)	11/19	9/17	6/15
lotkin	4/11	4/11	2/9
phillips	6/22	4/22	5/16
shaw	8/14	7/13	4/12

EXPERIMENT 4.2. We turn to the solution of large-scale problems using Algorithms 2 and 3. Table 4.8 shows the truncation index, k, for LSQR, as well as the total number of bidiagonalization steps, ℓ, required by Algorithms 2 and 3. Each value is the rounded average over 20 test problems: 10 problems with different error vectors $\mathbf{e}$ that model white Gaussian noise for each one of the matrices $A \in \mathbb{R}^{500\times 500}$ and $A \in \mathbb{R}^{1000\times 1000}$. By the design of the Algorithms 2 and 3, we have $\ell \geq k+4$; however, ℓ may be larger. Table 4.8 indicates that ℓ typically is not much larger than k.

Table 4.9 is analogous to Table 4.6 and uses the factors 2 and 5 in formulas similar to (4.4) and (4.5). Square matrices $A \in \mathbb{R}^{n\times n}$ and rectangular matrices $A \in \mathbb{R}^{2n\times n}$, with $n = 500$ or $n = 1000$, are used for each problem type. Every linear system of equations is solved for 10 different noise vectors $\mathbf{e}$. Our comparison includes the L-triangle method [8] associated with an L-curve approach, the Quadrature method described in [18], and the Ratio method introduced in [24].

COSE produces very good results for consistent linear systems, both square and rectangular ones. The results are less impressive for inconsistent problems. Nevertheless, only the quasi-optimality criterion exhibits a better performance for inconsistent least-squares problems.

Our next example is very large and very ill-conditioned. The matrix $A \in \mathbb{R}^{n\times n}$ is the prolate matrix described in [28] with $n = 100,000$. This is a Toeplitz matrix with entries

$$a_{ii} = 2\,\omega, \qquad a_{ij} = \frac{\sin(2\pi\,|i-j|\,\omega)}{\pi\,|i-j|} \quad \text{for } i \neq j,$$

with $\omega = 1/4$. The condition number estimator in MATLAB yields that the condition number $\|A\|\,\|A^{\dagger}\|$ of A is approximately 10^{17} when $n = 100$. We use the MATLAB toolbox `smt` [22] for handling A. This toolbox stores A in an optimized format and provides fast algorithms for matrix operations. Matrix-vector products are evaluated with the aid of the fast Fourier transform.

TABLE 4.9
Parameter choice rules for LSQR.

Method	Square systems	Overdetermined systems $\xi = 0$	$\xi = 1$	$\xi = 10$
COSE	3% (0%)	3% (0%)	24% (5%)	46% (28%)
L-corner	17% (4%)	19% (3%)	41% (22%)	68% (49%)
L-triangle	24% (9%)	20% (5%)	53% (28%)	77% (58%)
Res L-curve	51% (13%)	61% (16%)	77% (48%)	89% (73%)
Cond L-curve	63% (11%)	65% (28%)	74% (47%)	77% (58%)
Regińska	22% (5%)	33% (8%)	70% (36%)	82% (62%)
ResReg	22% (5%)	33% (8%)	70% (36%)	82% (61%)
Quasiopt	14% (2%)	14% (2%)	14% (2%)	18% (6%)
Extrapolation	71% (25%)	87% (56%)	91% (73%)	89% (71%)
Quadrature	54% (11%)	70% (22%)	88% (52%)	89% (68%)
Ratio	21% (0%)	28% (1%)	67% (31%)	80% (49%)
Discrepancy	48% (5%)	61% (12%)	71% (30%)	75% (42%)

TABLE 4.10
A very large scale example: prolate *matrix of dimension* $n = 100,000$.

Noise level	k_{best}	Error	k_{est}	Error	Time
10^{-4}	15	$7.47 \cdot 10^{-5}$	16	$7.47 \cdot 10^{-5}$	3.21
10^{-3}	8	$7.09 \cdot 10^{-4}$	10	$7.09 \cdot 10^{-4}$	4.27
10^{-2}	6	$7.07 \cdot 10^{-3}$	5	$7.07 \cdot 10^{-3}$	2.23
10^{-1}	1	$7.06 \cdot 10^{-2}$	1	$7.06 \cdot 10^{-2}$	2.86

The second and third columns of Table 4.10 report the LSQR iteration k_{best}, which yields the iterate closet to $\widehat{\mathbf{x}}$ in the Euclidean norm, as well as the relative error $\|\mathbf{x}_{k_{\text{best}}} - \widehat{\mathbf{x}}\|/\|\mathbf{x}_{k_{\text{best}}}\|$, for four noise levels. Columns four and five display the termination index, k_{est}, determined by Algorithms 2–3, and the relative error $\|\mathbf{x}_{k_{\text{est}}} - \widehat{\mathbf{x}}\|/\|\mathbf{x}_{k_{\text{est}}}\|$. Comparing columns two and four shows k_{est} to be very close to k_{best} for all noise levels. Moreover, $\mathbf{x}_{k_{\text{est}}}$ is seen to approximate $\widehat{\mathbf{x}}$ as well as $\mathbf{x}_{k_{\text{best}}}$ for all noise levels. The last column contains the execution time in seconds required by our method on an Intel Core i7-860 computer, with 8 Gb RAM, running Linux and MATLAB version 8.1. The regularized approximate solutions determined for the noise levels $\nu = 10^{-2}$ and $\nu = 10^{-1}$ are displayed in Figure 4.1 together with the exact solution $\widehat{\mathbf{x}}$. Our method is able to determine the LSQR truncation parameter accurately and fairly quickly.

5. Two case studies. To ascertain the effectiveness of our method when applied to the solution of real-world linear discrete ill-posed problems, we consider two applications, image deblurring and an inverse problem in geophysics, starting with the former. Consider the test images depicted in Figure 5.1. The blurred image $\mathbf{b} = [b_{i_1,i_2}]$, $i_1, i_2 = 1, \ldots, n$, is obtained by the convolution

$$b_{i_1,i_2} = \sum_{j_1,j_2=1}^{n} a_{i_1-j_1,i_2-j_2} x_{j_1,j_2},$$

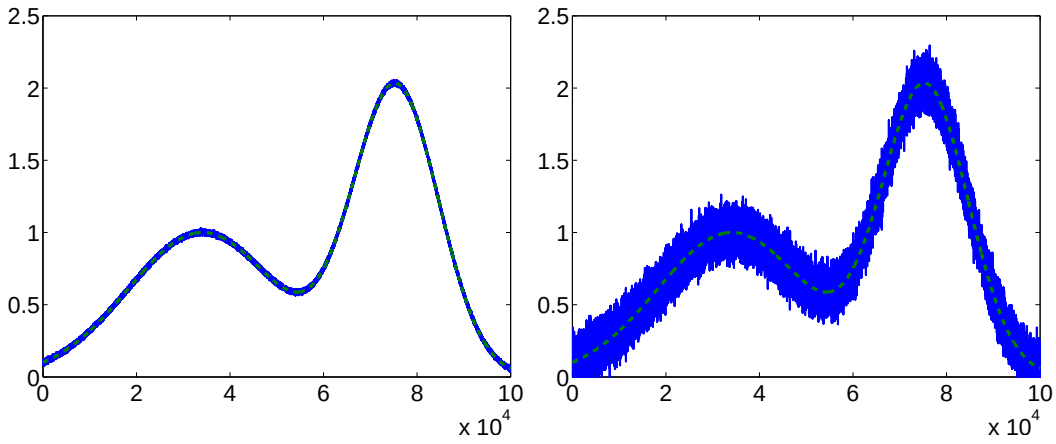

FIG. 4.1. *A very large scale example:* prolate *matrix of dimension* $n = 100,000$. *On the left-hand side is the approximate solution obtained with noise level* $\nu = 10^{-2}$, *the right-hand side shows the approximate solution for noise level* 10^{-1}. *The dashed graph depicts* $\widehat{\mathbf{x}}$.

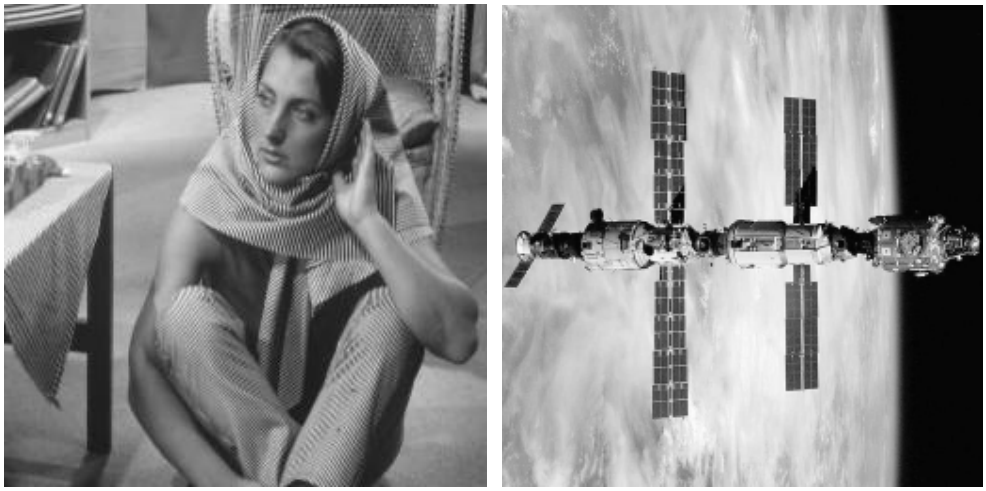

FIG. 5.1. *Test images:* BARBARA *on the left and* SPACESTAT *on the right.*

where $\mathbf{x} = [x_{j_1,j_2}]$ represents the original unblurred image and $A = [a_{k_1,k_2}]$ is the so-called *point spread function* (PSF). The latter is defined by Gaussian blur, i.e., the entries are of the form

$$a_{k_1,k_2} = \sqrt{\frac{\det(\Sigma)}{2\pi}} \cdot \exp\left(-\frac{\mathbf{k}^T\Sigma\mathbf{k}}{2}\right), \quad \mathbf{k} = [k_1, k_2]^T,$$

where

$$\Sigma = \begin{bmatrix} \rho_1 & \theta \\ \theta & \rho_2 \end{bmatrix}$$

is a positive definite parameter matrix. In our tests $\rho_1 = \rho_2 = 0.4, 0.2, 0.1$ and $\theta = 0$. The linear system $A\mathbf{x} = \mathbf{b}$ obtained after reordering the entries of the images columnwise has a coefficient matrix which is symmetric positive definite and block Toeplitz with Toeplitz blocks. This kind of matrices are of great importance in image

TABLE 5.1

Results obtained by deblurring the BARBARA *and* SPACESTAT *test images: ρ_1 is the blur parameter, ν the noise level, k_{best} the LSQR iteration which produces the smallest 2-norm error, while k_{est} and k_{lc} are the iterations identified by COSE and L-corner, respectively. Each value of k is followed by the corresponding error.*

Image	ρ_1	ν	k_{best}	Error	k_{est}	Error	k_{lc}	Error
BARBARA	0.4	10^{-2}	15	$1.15 \cdot 10^1$	26	$1.22 \cdot 10^1$	45	$1.58 \cdot 10^1$
		10^{-1}	3	$1.50 \cdot 10^1$	8	$2.45 \cdot 10^1$	9	$2.75 \cdot 10^1$
	0.2	10^{-2}	22	$1.27 \cdot 10^1$	34	$1.31 \cdot 10^1$	53	$1.49 \cdot 10^1$
		10^{-1}	4	$1.60 \cdot 10^1$	9	$2.04 \cdot 10^1$	8	$1.89 \cdot 10^1$
	0.1	10^{-2}	36	$1.43 \cdot 10^1$	41	$1.43 \cdot 10^1$	60	$1.50 \cdot 10^1$
		10^{-1}	6	$1.76 \cdot 10^1$	12	$2.10 \cdot 10^1$	8	$1.81 \cdot 10^1$
SPACESTAT	0.4	10^{-2}	15	$1.23 \cdot 10^1$	28	$1.37 \cdot 10^1$	45	$1.87 \cdot 10^1$
		10^{-1}	3	$1.77 \cdot 10^1$	7	$2.74 \cdot 10^1$	9	$3.52 \cdot 10^1$
	0.2	10^{-2}	22	$1.50 \cdot 10^1$	33	$1.54 \cdot 10^1$	52	$1.80 \cdot 10^1$
		10^{-1}	4	$1.90 \cdot 10^1$	9	$2.62 \cdot 10^1$	8	$2.40 \cdot 10^1$
	0.1	10^{-2}	30	$1.68 \cdot 10^1$	40	$1.70 \cdot 10^1$	59	$1.82 \cdot 10^1$
		10^{-1}	6	$2.10 \cdot 10^1$	12	$2.56 \cdot 10^1$	8	$2.17 \cdot 10^1$

restoration, because they are often used to model atmospheric blurring [16]. The size of our test images is $n = 256$, so the dimension of the system is $n^2 = 65536$.

It is straightforward to prove that, under the above assumption of $\theta = 0$, the block matrix A can be expressed as the tensor product of two Toeplitz matrices of size n. This allows to perform the matrix-vector product $A\mathbf{z}$, where $\mathbf{z}$ is the lexicographically reordered image, by multiplying the original image by the two tensor factors of A, which requires a smaller complexity. To handle the Toeplitz matrices and to perform fast matrix products we use the MATLAB toolbox `smt` [22]. After applying the Gaussian blur to each test image, we contaminate it with noise as in (4.1), with $\nu = 10^{-2}, 10^{-1}$.

Table 5.1 shows results obtained when applying Algorithm 2–3 to the deblurring problem. The LSQR iterate selected by COSE is denoted by k_{est}, while the iterate k_{best} has the smallest error in the Euclidean norm, and the iterate k_{lc} is selected by the L-corner method [17]. In each row of the table, the iteration number is followed by the corresponding error of the iterate in the Euclidean norm.

Figures 5.2 and 5.3 show the two blurred and noisy images obtained by setting $\rho_1 = \rho_2 = 0.2$. The noise level is $\nu = 10^{-2}$. The figures also show the restorations determined by COSE and the L-corner method, as well as the images corresponding to the iterates with the smallest Euclidean norm.

The L-corner method performs very well. COSE determines images that generally are of the same quality as those determined by the L-corner method, and sometimes yields images of slighty higher quality. An advantage of Algorithm 2–3 is that it requires only four additional iterations to stop after locating a local minimum, while the L-corner method generally needs a larger number of iterations to correctly identify the "corner" of an L-curve. We remark that, while the generalized cross validation (GCV) method [14] is a popular method for determining the regularization parameter, its application to large-scale problem is fairly complicated; see [12] for a discussion. We therefore do not compare with the performance of GCV for the present example.

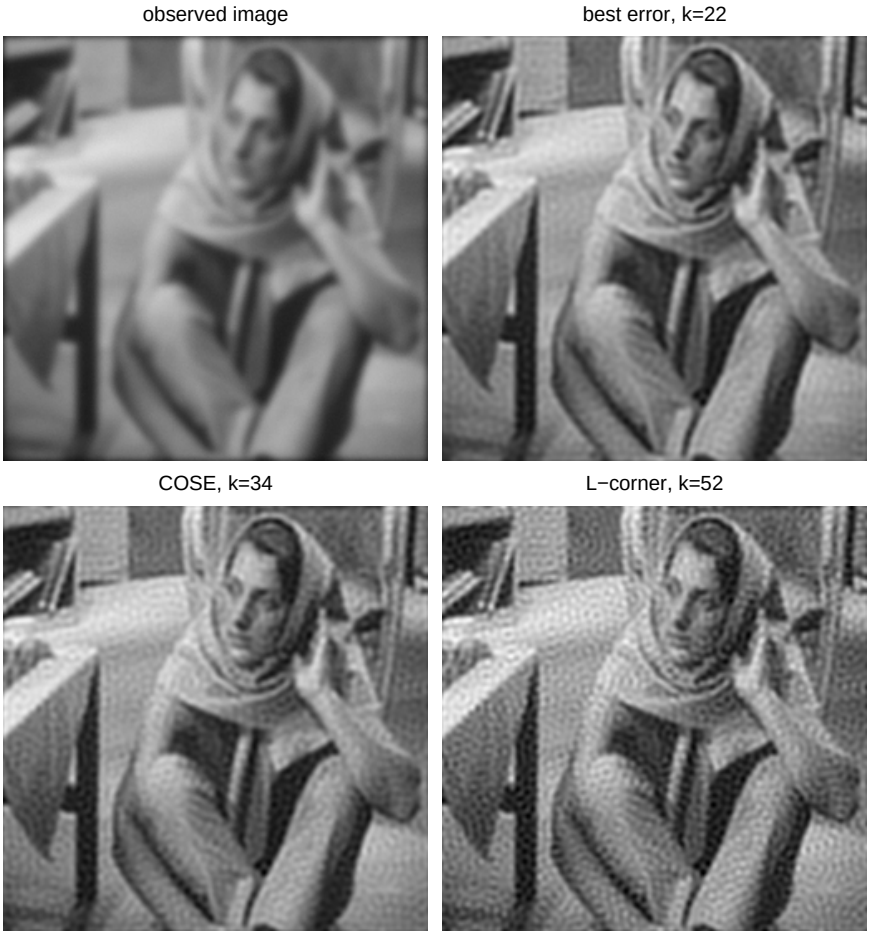

FIG. 5.2. *Deblurring of the* BARBARA *test image with* $\rho_1 = 0.2$, $\nu = 10^{-2}$*: blurred image (top left), minimum 2-norm error (top right), COSE method (bottom left), and L-corner method (bottom right).*

We turn to our second application. A ground conductivity meter is a device used for electromagnetic sounding. It is composed by two coils (a transmitter and a receiver) placed at the end points of a bar. The transmitter produces a primary magnetic field which induces small currents in the ground. These currents produce a secondary magnetic field that is sensed by the receiver coil. The device has two operating positions, which produce different measures, corresponding to the orientation (vertical or horizontal) of the electric dipole associated with the transmitter coil. The measurements can be taken at different heights above the ground and are influenced by the electrical conductivity of the underground layers.

A linear model for this device has been proposed in [20]. It is valid when the ground does not include substances with very large electrical conductivity and/or moderately large magnetic permeability. The model consists of the following pair of

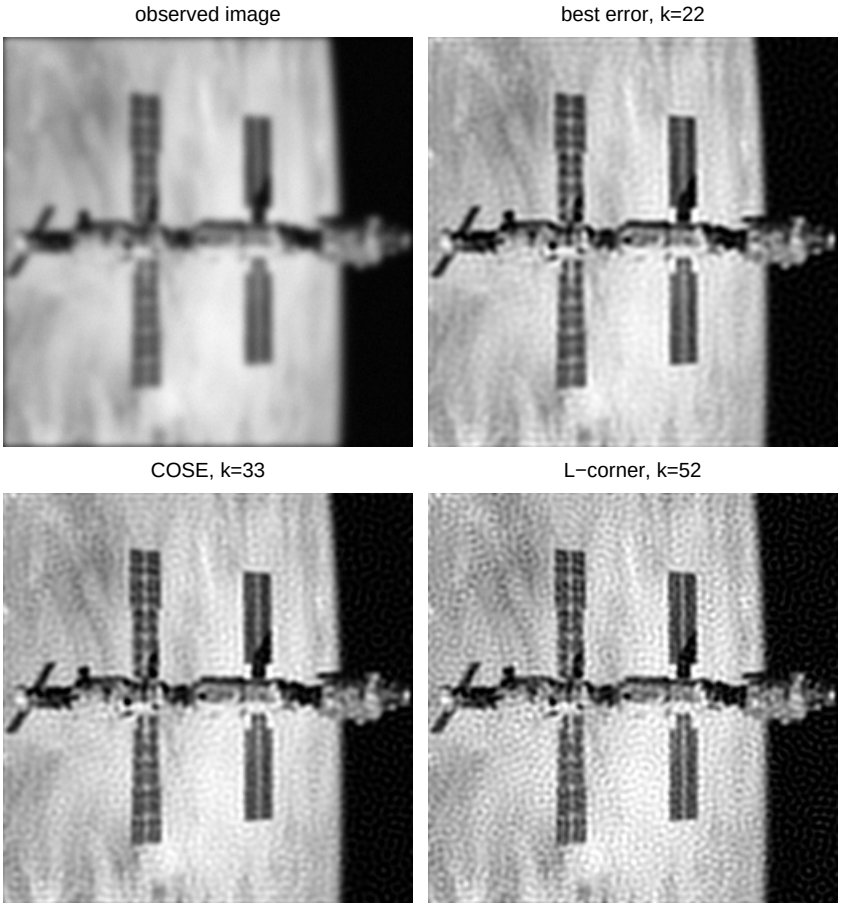

FIG. 5.3. *Deblurring of the* SPACESTAT *test image with* $\rho_1 = 0.2$, $\nu = 10^{-2}$*: blurred image (top left), minimum 2-norm error (top right), COSE method (bottom left), and L-corner method (bottom right).*

integral equations

$$m^V(h) = \int_0^\infty \phi^V(h+z)\sigma(z)\,dz,$$
$$m^H(h) = \int_0^\infty \phi^H(h+z)\sigma(z)\,dz,$$

which correspond to the vertical and horizontal position of the device, respectively, where

$$\phi^V(z) = \frac{4z}{(4z^2+1)^{3/2}}, \qquad \phi^H(z) = 2 - \frac{4z}{(4z^2+1)^{1/2}}.$$

The two equations relate the apparent conductivity $m(h)$, measured at height h over the ground, to the real conductivity $\sigma(z)$, with z being the ratio between the depth and the inter-coil distance. The conductivities are measured in millisiemens per meter (mS/m).

Following [4], we discretize the integral equations by a simple Galerkin method, which approximates the function $\sigma(z)$ by a linear combination of piecewise constant

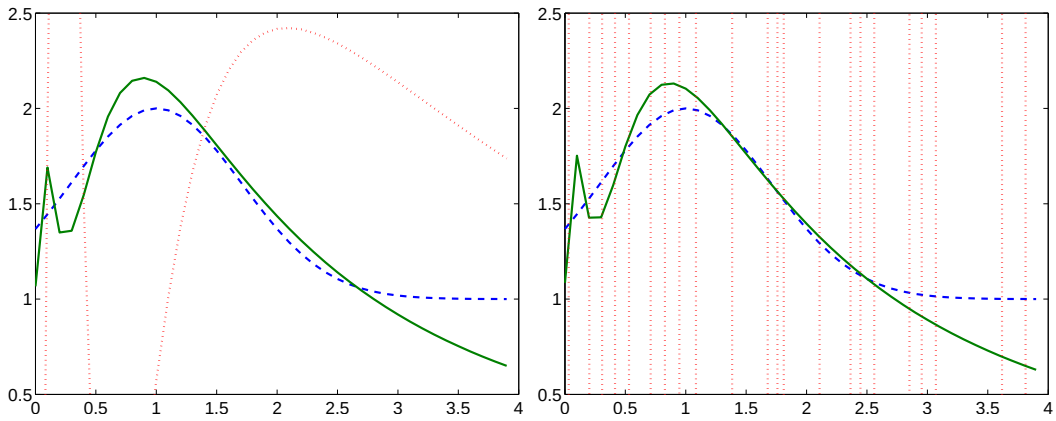

FIG. 5.4. *Solution of the ground conductivity problem with two different noise realizations: exact solution (dashed line), COSE solution (continuous line), and GCV solution (dotted line). The COSE method selected the TSVD solution with $k = 3$ in both tests; this value of k produces also the minimal 2-norm error. GCV selected $k = 4$ in the left graph and $k = 39$ in the right one.*

functions. This amounts to considering n layers of width δ below the surface of the ground, each characterized by constant electrical conductivity; we let $\delta = 0.1$ meters. The measurement heights were chosen to be $h_i = i\delta$, $i = 0, 1, \ldots, m-1$. The matrix of the resulting linear system of equations $K\boldsymbol{\sigma} = \mathbf{b}$ is of size $2m \times n$; the solution vector contains the value of the conductivity in each layer.

It has been observed experimentally that the noise level in the horizontal measurement position is different from the noise level in the vertical measurement position. We first compute the right-hand side $\mathbf{b}$ corresponding to a test solution and then perturb its first m components by (4.1) corresponding to 1% error and the remaining m components corresponding to 10% error. We let $m = 20$ and $n = 40$ in our computations. These values of the parameters are compatible with real data sets.

We applied Algorithm 1 to this problem and carried out 40 tests with different noise realizations. The COSE method identified the optimal solution, obtained for $k = 3$, 34 times. This solution localizes rather accurately the position of the maximum, a property of importance in this application, because it indicates the presence of a high conductive material at a certain depth. In 6 tests the COSE method produced an over-regularized solution, with $k = 1$, presenting only one relative maximum at a smaller depth.

The GCV method is less effective for this problem, probably because the noise is not equally distributed. GCV identified the optimal solution only 27 times (of 40), and in the remaining 13 experiments, GCV produced under-regularized solutions that are quickly oscillating. They cannot be used to identify the position of the maximum. Figure 5.4 shows two of the tests in which this behavior is particularly evident.

6. Conclusions. This paper describes a new simple approach to determine suitable values of the regularization parameters for Tikhonov regularization and TSVD when no estimate the norm of the noise in the data is available. An extension suitable to use with the LSQR method also is presented. The method also yields an estimate of the norm of the noise in the data. The numerical examples show the method to be competitive with other data-driven methods for determining regularization parameters. Moreover, the method typically yields a quite accurate estimate of the noise

level.

Acknowledgment. The authors would like to thank Gian Piero Deidda (Department of Land Engineering, University of Cagliari) for fruitful discussions about the inverse problem in Geophysics discussed in Section 5.

REFERENCES

[1] M. L. Baart, *The use of auto-correlation for pseudo-rank determination in noisy ill-conditioned least-squares problems*, IMA J. Numer. Anal., 2 (1982), pp. 241–247.
[2] J. Baglama and L. Reichel, *Augmented implicitly restarted Lanczos bidiagonalization methods*, SIAM J. Sci. Comput., 27 (2005), pp. 19–42.
[3] F. Bauer and M. A. Lukas, *Comparing parameter choice methods for regularization of ill-posed problem*, Math. Comput. Simulation, 81 (2011), pp. 1795–1841.
[4] B. Borchers, T. Uram, and J. M. H. Hendrickx, *Tikhonov regularization of electrical conductivity depth profiles in field soils*, Soil Sci. Soc. Am. J., 61 (1997), pp. 1004–1009.
[5] C. Brezinski, G. Rodriguez, and S. Seatzu, *Error estimates for the regularization of least squares problems*, Numer. Algorithms, 51 (2009), pp. 61–76.
[6] D. Calvetti, G. H. Golub, and L Reichel, *Estimation of the L-curve via Lanczos bidiagonalization*, BIT, 39 (1999), pp. 603–619.
[7] D. Calvetti, B. Lewis, and L Reichel, *GMRES, L-curves, and discrete ill-posed problems*, BIT, 42 (2002), pp. 44–65.
[8] J. L. Castellanos, S. Gómez, and V. Guerra, *The triangle method for finding the corner of the L-curve*, Appl. Numer. Math., 43 (2002), pp. 359–373.
[9] L. M. Delves and J. L. Mohammed, *Computational Methods for Integral Equations*, Cambridge University Press, Cambridge, 1985.
[10] H. W. Engl, M. Hanke, and A. Neubauer, *Regularization of Inverse Problems*, Kluwer, Dordrecht, 1996.
[11] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed., Johns Hopkins University Press, Baltimore, 1996.
[12] G. H. Golub and U. von Matt, *Generalized cross-validation for large-scale problems*, J. Comp. Graph. Stat., 6 (1997), pp. 1–34.
[13] P. C. Hansen, *Analysis of the discrete ill-posed problems by means of the L-curve*, SIAM Review, 34 (1992), pp. 561–580.
[14] ———, *Rank-Deficient and Discrete Ill-Posed Problems*, SIAM, Philadelphia, 1998.
[15] ———, *Deconvolution and regularization with Toeplitz matrices*, Numer. Algorithms 46 (2002), pp. 323–378.
[16] ———, *Regularization tools version 4.0 for MATLAB 7.3*, Numer. Algorithms, 46 (2007), pp. 189–194.
[17] P. C. Hansen, T. K. Jensen, and G. Rodriguez, *An adaptive pruning algorithm for the discrete L-curve criterion*, J. Comput. Appl. Math., 198 (2006), pp. 483–492.
[18] I. Hnětynková, M. Plešinger, and Z. Strakoš, *The regularizing effect of the Golub–Kahan iterative bidiagonalization and revealing the noise level in the data*, BIT, 49 (2009), pp. 669–696.
[19] S. Kindermann, *Convergence analysis of minimization-based noise-level-free parameter choice rules for ill-posed problems*, Electron. Trans. Numer. Anal., 38 (2011), pp. 233–257.
[20] J. D. McNeill, *Electromagnetic terrain conductivity measurement at low induction numbers*, Technical Report TN-6, Geonics Limited, Mississauga, Ontario, Canada, 1980.
[21] C. C. Paige and M. A. Saunders, *LSQR: An algorithm for sparse linear equations and sparse least squares*, ACM Trans. Math. Software, 8 (1982), pp. 43–71.
[22] M. Redivo-Zaglia and G. Rodriguez, *smt: a Matlab toolbox for structured matrices*, Numer. Algorithms, 59 (2012), pp. 639–659.
[23] T. Regińska, *A regularization parameter in discrete ill-posed problems*, SIAM J. Sci. Comput., 17 (1996), pp. 740–749.
[24] L. Reichel and G. Rodriguez, *Old and new parameter choice rules for discrete ill-posed problems*, Numer. Algorithms, 63 (2013), pp. 65–87.
[25] L. Reichel, G. Rodriguez, and S. Seatzu, *Error estimates for large-scale ill-posed problems*, Numer. Algorithms, 51 (2009), pp. 341–361.

[26] L. Reichel and H. Sadok, *A new L-curve for ill-posed problems*, J. Comput. Appl. Math., 219 (2008), pp. 493–508.
[27] J. M. Varah, *Pitfalls in the numerical solution of linear ill-posed problems*, SIAM J. Sci. Statist. Comput., 4 (1983), pp. 164–176
[28] ———, *The prolate matrix*, Linear Algebra Appl., 187 (1993), pp. 269–278.