Large-Scale Tikhonov Regularization via Reduction by Orthogonal Projection

Jörg Lampe * Lothar Reichel † Heinrich Voss ‡

Dedicated to Danny C. Sorensen on the occasion of his 65th birthday.

Keywords: least squares, general-form Tikhonov regularization, discrepancy principle, ill-posedness

AMS Subject Classification: 65F15, 65F22, 65F30

Abstract

This paper presents a new approach to computing an approximate solution of Tikhonov-regularized large-scale ill-posed least-squares problems with a general regularization matrix. The iterative method applies a sequence of projections onto generalized Krylov subspaces. A suitable value of the regularization parameter is determined by the discrepancy principle.

1 Introduction

This paper is concerned with the computation of an approximate solution of least-squares problems of the form

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

with a matrix $A \in \mathbb{R}^{m\times n}$, $m \geq n$, of ill-determined rank. In particular, A is severely ill-conditioned and possibly singular. Matrices of this kind arise, for instance, from the discretization of ill-posed problems, such as Fredholm integral equations of the first kind; see, e.g., [10, 16] for discussions. The vector $b \in \mathbb{R}^m$ represents observations and is assumed to be contaminated by an error $e \in \mathbb{R}^m$, which may stem from measurement inaccuracies. Throughout this paper $\|\cdot\|$ denotes the Euclidean vector norm or the associated induced matrix norm.

*Institute of Numerical Simulation, Hamburg University of Technology, D-21071 Hamburg, Germany (joerg.lampe@tu-harburg.de)

†Department of Mathematical Sciences, Kent State University, Kent, OH 44242, USA (reichel@math.kent.edu)

‡Institute of Numerical Simulation, Hamburg University of Technology, D-21071 Hamburg, Germany (voss@tu-harburg.de)

Let b_{true} denote the unknown error-free vector associated with b, i.e.,

$$b = b_{\text{true}} + e. \tag{1.2}$$

The linear system of equations $Ax = b_{\text{true}}$ associated with the least-squares problem (1.1) is assumed to be consistent. We would like to determine an approximation of its solution of minimal Euclidean norm, x_{true}, by computing a suitable approximate solution of (1.1).

Due to the ill-conditioning of A and the error e in b, straightforward solution of (1.1) often does not yield a meaningful approximation of x_{true}. It is necessary to stabilize the computations by regularization. One of the most popular regularization methods is due to Tikhonov. This method replaces (1.1) by a penalized least-squares problem of the form

$$\min_{x\in\mathbb{R}^n} \left\{ \|Ax - b\|^2 + \mu^{-1}\|Lx\|^2 \right\}. \tag{1.3}$$

The scalar $\mu \in (0, \infty)$ is referred to as the regularization parameter and $L \in \mathbb{R}^{p\times n}$ as the regularization matrix. This matrix defines a (semi-)norm, $\|L \cdot \|$, on the solution space. The normal equations associated with (1.3) are given by

$$(A^T A + \mu^{-1} L^T L)x = A^T b. \tag{1.4}$$

They have the unique solution

$$x_\mu = (A^T A + \mu^{-1} L^T L)^{-1} A^T b \tag{1.5}$$

for any $\mu > 0$ when

$$\operatorname{rank} \begin{bmatrix} A \\ L \end{bmatrix} = n. \tag{1.6}$$

We assume this to be the case. The use of μ^{-1} above, instead of μ, is commented on in Subsection 2.2.

When the matrices A and L are small, solutions x_μ of (1.3) easily can be determined for many values of $\mu > 0$ by first computing the generalized singular value decomposition (GSVD) of the matrix pair $\{A, L\}$. For large-scale problems and a fixed $\mu > 0$, an approximation of x_μ can be determined by applying an iterative method, such as LSQR, to (1.4). However, generally, a suitable value of the parameter μ is not known a priori and has to be determined during the solution process. Many approaches to determining an appropriate value of μ, including the L-curve criterion [15, 16, 22], the discrepancy principle [10, 23], generalized cross validation [7, 11], and information criteria [1, 28], require the normal equations (1.4) to be solved repeatedly for many different values of the parameter μ. This can make application of LSQR costly.

When $L = I$, the Tikhonov regularization problem (1.3) is said to be in *standard form*. Approximations of the solution x_μ of problems in standard form can be computed by partial Lanczos bidiagonalization of A; see, e.g., [2, 3, 4, 5, 6, 13, 14] for several solution methods based on this approach. The computed approximation, x_μ^ℓ, lives in the Krylov subspace

$$\mathcal{K}_\ell(A^TA, A^Tb) = \text{span}\{A^Tb, (A^TA)A^Tb, \ldots, (A^TA)^{\ell-1}A^Tb\} \tag{1.7}$$

for some $\ell \geq 1$. Due to the shift invariance of Krylov subspaces, i.e., the property that

$$\mathcal{K}_\ell(A^TA, A^Tb) = \mathcal{K}_\ell(A^TA + \mu^{-1}I, A^Tb), \qquad \mu > 0,$$

the spaces (1.7) can be used for all values of $\mu > 0$. This makes it possible to first project the problem (1.4) onto a Krylov subspace (1.7) and then regularize the projected problem by Tikhonov's method. This approach is used in many of the already mentioned references. The first hybrid method of this kind was proposed in [24].

Solution by partial Lanczos bidiagonalization also can be applied to Tikhonov regularization problems (1.3) with $L \neq I$, provided that the regularization problem can be transformed to standard form without too much effort. This transformation is carried out with the aid of the substitutions $y = Lx$ and $x = L_A^\dagger y$, where, when $p \leq n$,

$$L_A^\dagger := (I - (A(I - L^\dagger L))^\dagger A)L^\dagger.$$

This matrix is referred to as the *A-weighted pseudoinverse of L*; see [9] for a discussion of its properties. When L is banded with small bandwidth and has a known null space, transformation of (1.3) to standard form is attractive. Other situations when transformation to standard form is feasible are discussed in [27]. However, when (1.3) is the discretization of an integral equation of the first kind in two or more space-dimensions, the regularization matrix L often is chosen to be a sum of Kronecker products. The expression for $L_A^\dagger$ then is complicated and unattractive to use; see Example 4.2 of Section 4 for an illustration.

Kilmer et al. [19] recently proposed a projection method for large-scale Tikhonov-regularized least-squares problems that are infeasible or expensive to transform to standard form. The method computes an approximation of a partial GSVD of the matrix pair $\{A, L\}$ by an inner-outer iteration scheme. An attractive property of this method is that the computed approximate partial GSVD is independent of μ. However, the inner-outer iteration scheme may be expensive, due to the possibly fairly large number of required matrix-vector product evaluations with A and its transpose A^T. We therefore are interested in developing an alternative approach.

A scheme that projects L into a Krylov subspace (1.7) and determines an approximate solution of (1.3) in this subspace is described in [18]. The approach of the present paper differs in that the subspace in which we determine an approximate solution of (1.3) depends on L. A method based on reducing both A and L by an Arnoldi-type method is presented in [25]. This method requires both A and L to be square matrices, but we note that this restriction in some situations can be overcome, e.g., by zero-padding. Our approach allows both A and L to be general rectangular matrices.

This paper proposes an iterative projection method that computes an approximate solution of (1.5) in a generalized Krylov subspace. The regularization parameter is determined by the *discrepancy principle*. We assume that an estimate of the norm of the error e in the vector b in (1.1) is available,

$$\varepsilon \approx \|e\|.$$

The regularization parameter $\mu = \mu(\varepsilon)$ is chosen so that the computed approximation $\widetilde{x}_\mu$ of the solution x_μ of (1.3) satisfies

$$\|A\widetilde{x}_\mu - b\| = \eta\varepsilon =: \delta, \tag{1.8}$$

where $\eta > 1$ is a user-specified constant, whose size depends on the accuracy of the estimate ε of $\|e\|$. The more accurate the estimate is, the closer we let η be to one. A vector $\widetilde{x}_\mu$, such that (1.8) holds, is said to satisfy the discrepancy principle.

Introduce the function

$$\varphi(\mu) := \|Ax_\mu - b\|^2, \tag{1.9}$$

where x_μ is given by (1.5), and let $\bar{\mu}$ satisfy

$$\varphi(\bar{\mu}) = \delta^2. \tag{1.10}$$

The function $\varphi(\mu)$ is convex and monotone. This follows from results in Subsection 2.2. A numerical method for inexpensively computing upper and lower bounds for $\bar{\mu}$ when $L = I$ is described in [6]. Note that the evaluation of $\varphi(\mu)$ is expensive when A is large. The inexpensive computation of bounds for $\bar{\mu}$, based on the connection between partial Lanczos bidiagonalization and Gaussian quadrature, as described in [6], therefore is attractive for large matrices A.

Assume that L is a regularization matrix, such that the computation with $L_A^\dagger$ is unattractive. Introduce the subspace $\mathcal{V} \subset \mathbb{R}^n$ of small dimension $k \ll n$, and let the columns of $V \in \mathbb{R}^{n\times k}$ form an orthonormal basis for $\mathcal{V}$. We propose to approximate $\varphi(\mu)$ by the function

$$\varphi(\mu; V) := \|Ax_\mu^k - b\|^2, \tag{1.11}$$

where the x_μ^k is obtained by solving the Tikhonov problem (1.3) restricted to $\mathcal{V} \subset \mathbb{R}^n$. Specifically, let

$$y_\mu^k = \operatorname{argmin}_{y\in\mathbb{R}^k} \left\{\|AVy - b\|^2 + \mu_k^{-1}\|LVy\|^2\right\}, \quad x_\mu^k = Vy_\mu^k. \tag{1.12}$$

The regularization parameter μ_k is determined as the zero of the function

$$f(\mu; V) := \|Ax_\mu^k - b\|^2 - \delta^2.$$

This zero can be computed, e.g., by Newton's method, by rational inverse interpolation (see [20, 21]), or by a cubically convergent zero-finder [26]. Having determined μ_k, the search space $\mathcal{V}$ is expanded by the gradient of the functional in (1.3) evaluated at x_μ^k. After expansion, a new value, μ_{k+1}, of the regularization parameter is calculated. Zero-finding and expansion is repeated until $\mu = \mu_k$ and the corresponding approximation $x_\mu^k = V y_\mu^k$ satisfies a stopping criterion; see Section 2. Since the expansion direction depends on the value μ_k of μ, the search space $\mathcal{V}$ is not a Krylov subspace. Numerical examples illustrate that the stopping criterion typically is satisfied for search spaces $\mathcal{V}$ of fairly small dimension.

The most expensive part of the proposed scheme is the repeated enlargement of the search space $\mathcal{V}$. Since application of the zero-finder to the projected problem is inexpensive, we enlarge $\mathcal{V}$ by a single vector as soon as a new value of the regularization parameter has been computed. The overall complexity of the presented method is $\mathcal{O}(mn)$ arithmetic floating-point operations, i.e., the complexity is of the same order as a matrix-vector product evaluation with the matrix A, when A is dense and unstructured.

This paper is organized as follows. Section 2 presents our new scheme for solving the Tikhonov regularization problem (1.3). Implementation issues are discussed in Section 3, and Section 4 contains numerical examples. They illustrate the efficiency of the presented approach. Concluding remarks can be found in Section 5.

2 Tikhonov Regularization via Orthogonal Projection

Let the function $\varphi(\mu)$ be defined by (1.9) with x_μ given by (1.5), and let δ be determined by (1.8). We assume that δ satisfies the inequalities

$$\|P_{\mathcal{N}(A^T)} b\| < \delta < \|b\|, \tag{2.1}$$

where $P_{\mathcal{N}(A^T)}$ denotes the orthogonal projector onto the null space $\mathcal{N}(A^T)$ of A^T. The conditions (2.1) on the error bound δ are natural for the Tikhonov least-squares problem (1.3); see Subsection 3.2 for details.

The computation of a value $\bar\mu$ of the regularization parameter, such that $\varphi(\bar\mu) = \delta^2$, is a nonlinear problem. The evaluation of $\varphi(\mu)$ for a given μ is rather expensive, because it requires the solution of the (large) Tikhonov least-squares problem (1.3). We therefore seek to determine approximations of $\bar\mu$ and of the corresponding Tikhonov solution $x_{\bar\mu} = (A^T A + \bar\mu L^T L)^{-1} A^T b$ by an iterative projection method, that replaces the original large problem (1.3) by a sequence of problems with search spaces of much smaller dimensions, and only solves the projected problems. Let $\mathcal{V} \subset \mathbb{R}^n$ be a subspace of dimension $k \ll n$, and let the columns of the matrix $V \in \mathbb{R}^{n\times k}$ form an orthonormal basis for $\mathcal{V}$. Introduce the matrix

$$T(\mu) := A^T A + \mu^{-1} L^T L \tag{2.2}$$

and let $y_\mu \in \mathbb{R}^k$ be given by

$$y_\mu = (V^T T(\mu) V)^{-1} V^T A^T b; \tag{2.3}$$

cf. (1.12). We approximate $\varphi(\mu)$ by the function

$$\varphi(\mu; V) := \|AVy_\mu - b\|^2, \tag{2.4}$$

with y_μ defined by (2.3), and determine $\mu = \mu_k$ so that $\varphi(\mu_k; V) = \delta^2$. We then use $x_{\mu_k} = Vy_{\mu_k}$ as an approximation of $x_{\bar{\mu}}$. When the dimension of $\mathcal{V}$ is n, we have

$$\varphi(\mu; V) = \varphi(\mu).$$

Experience with numerous computed examples, some of which are reported in Section 4, indicates that the dimension of $\mathcal{V}$ generally can be chosen fairly small.

2.1 Enlarging the Search Space

The choice of subspace is important for projection methods. We discuss this choice in the present and following subsections. Let μ_k be an available approximation of $\bar{\mu}$ and let the orthonormal columns of $V \in \mathbb{R}^{n \times k}$ span the current search space $\mathcal{V}$. The projection of the normal equations (1.4) into $\mathcal{V}$ can be expressed as

$$V^T T(\mu_k) V y = V^T A^T b, \tag{2.5}$$

where $T(\mu_k)$ is defined by (2.2). Let y_{μ_k} denote the minimal-norm solution of (2.5). Mapping y_{μ_k} back into $\mathbb{R}^n$, we obtain the vector $x_{\mu_k} = Vy_{\mu_k}$. The residual associated with (2.5) is given by

$$r_{\mu_k} = T(\mu_k) x_{\mu_k} - A^T b.$$

In the absence of round-off errors, the residual r_{μ_k} is orthogonal to the search space $\mathcal{V}$. To enforce orthogonality in the presence of round-off errors, we reorthogonalize r_{μ_k} against $\mathcal{V}$ and then include the reorthogonalized vector in $\mathcal{V}$. Thus, we let

$$\tilde{r}_{\mu_k} := (I - VV^T) r_{\mu_k}, \qquad v_{\text{new}} := \tilde{r}_{\mu_k} / \|\tilde{r}_{\mu_k}\|, \qquad V_{\text{new}} := [V, v_{\text{new}}].$$

We refer to the computations described as an *expansion step* for the search space $\mathcal{V}$.

The spaces $\mathcal{V}$ are Krylov subspaces only in special situations. If the current search space $\mathcal{V}$ equals the Krylov subspace $\mathcal{K}_k(T(\mu_k), A^T b)$, then the expanded search space also is a Krylov subspace. Specifically, the columns of V_{new} form an orthonormal basis for the Krylov subspace $\mathcal{K}_{k+1}(T(\mu_k), A^T b)$. The connection between this Krylov subspace and the Lanczos process shows that the matrix $V_{\text{new}}^T T(\mu_k) V_{\text{new}}$ is symmetric and tridiagonal. However, since the μ_k are updated during the computations, so is $T(\mu_k)$. Therefore, the space $\mathcal{V}$, in general, is not a Krylov subspace when $L \neq I$. In particular, the new basis vector v_{new} cannot be computed with a short recurrence relation. We refer to the search spaces $\mathcal{V}$ as *generalized Krylov subspaces*.

The cost of enlarging the search space by one dimension is $\mathcal{O}(mn)$ arithmetic floating point operations, and so is the multiplication of the matrix $T(\mu_k)$ by a vector; see also Subsection 3.1. This cost is higher than the determination of a suitable Tikhonov parameter for a projected problem. We therefore compute a new value of the regularization parameter after each increase of $\dim(\mathcal{V})$ by one.

2.2 Computation of the Regularization Parameter

Let the search space $\mathcal{V}$ of dimension k be available, and let the columns of $V \in \mathbb{R}^{n\times k}$ form an orthonormal basis. It is quite natural to let the regularization parameter μ solve the nonlinear equation

$$\varphi(\mu; V) = \delta^2 \tag{2.6}$$

with φ defined by (2.4) and δ by (1.8). This choice of μ secures that the discrepancy principle holds for the current search space $\mathcal{V}$. It is convenient to consider equation (2.6) a zero-finding problem for the function

$$f(\mu; V) := \varphi(\mu; V) - \delta^2. \tag{2.7}$$

This subsection discusses some properties of the function $\varphi(\mu; V)$. We show that $\varphi(\mu; V)$ is monotonically decreasing and convex in the interval $\mu \in (0,\infty)$ under a mild condition on the search space $\mathcal{V}$. Let the matrix $V \in \mathbb{R}^{n\times k}$ be fixed. Then the function $\varphi(\mu; V)$ for $\mu > 0$ can be expressed as

$$\begin{aligned} \varphi(\mu; V) &= \|AV(V^T T(\mu)V)^{-1}V^T A^T b - b\|^2 \\ &= \|AV(V^T(A^TA + \mu^{-1}L^TL)V)^{-1}V^TA^Tb - b\|^2. \end{aligned} \tag{2.8}$$

Theoretical results can be obtained by introducing the GSVD of the matrix pair $\{AV, LV\}$, with $AV \in \mathbb{R}^{m\times k}$ and $LV \in \mathbb{R}^{p\times k}$,

$$AV = UCX^T, \tag{2.9}$$

$$LV = \hat{V}SX^T, \tag{2.10}$$

where the matrices $U = [u_1, u_2, \ldots, u_k] \in \mathbb{R}^{m\times k}$ and $\hat{V} \in \mathbb{R}^{p\times k}$ have orthonormal columns, and the entries of the matrices $C = \mathrm{diag}[c_1, c_2, \ldots, c_k] \in \mathbb{R}^{k\times k}$ and $S = \mathrm{diag}[s_1, s_2, \ldots, s_k] \in \mathbb{R}^{k\times k}$ are ordered according to $0 \leq c_1 \leq \ldots \leq c_k \leq 1$ and $1 \geq s_1 \geq \ldots \geq s_k \geq 0$ with $c_j^2 + s_j^2 = 1$ for $1 \leq j \leq k$. The c_j and s_j are referred to as generalized singular values. Finally, the matrix $X \in \mathbb{R}^{k\times k}$ is nonsingular; see, e.g., [12, Section 8.7.2] for details on the GSVD. In our application, we have $m \geq n \geq p \gg k$. We use the GSVD as an analytical tool, whereas the numerical method described in Section 3 uses computationally less demanding QR factorizations.

Substituting the decompositions (2.9) and (2.10) into (2.8) gives us a more tractable expression. Using the relations

$$V^TA^TAV = XC^2X^T, \quad V^TL^TLV = XS^2X^T, \quad V^TA^Tb = XCU^Tb,$$

we obtain

$$\begin{aligned}\varphi(\mu; V) &= \|UCX^T(XC^2X^T + \mu^{-1}XS^2X^T)^{-1}XCU^Tb - b\|^2 \\ &= \|UC(C^2 + \mu^{-1}S^2)^{-1}CU^Tb - b\|^2 \\ &= b^Tb + b^TUC\left((C^2+\mu^{-1}S^2)^{-2} - 2(C^2+\mu^{-1}S^2)^{-1}\right)CU^Tb \\ &= b^Tb + \sum_{i=1}^{k}\left(\frac{s_i^4}{(\mu c_i^2 + s_i^2)^2} - 1\right)|u_i^Tb|^2, \end{aligned} \tag{2.11}$$

which yields the following formulas for the derivative of φ with respect to μ:

$$\varphi'(\mu; V) = -\sum_{i=1}^{k}\frac{2c_i^2s_i^4}{(\mu c_i^2 + s_i^2)^3}|u_i^Tb|^2 = -\sum_{s_i>0}\frac{2\gamma_i^2}{(1+\mu\gamma_i^2)^3}|u_i^Tb|^2 \le 0, \tag{2.12}$$

where $\gamma_i = c_i/s_i$. Note that all terms in (2.11) and (2.12) are well defined since $c_i + s_i > 0$ for all i. This follows from the fact that the matrix $[A^T, L^T]^TV$ is of full rank, which is a consequence of (1.6).

Here and below we assume that the search space $\mathcal{V}$ is such that

$$b_U \notin \mathcal{N}(LVX^{-T}) \quad \text{with} \quad b_U = X^{-1}V^TA^Tb. \tag{2.13}$$

The vector b_U can be expressed as $b_U = CU^Tb$ with entries $b_U(i) = c_i(u_i^Tb)$. Moreover, in view of that $\mathcal{N}(LVX^{-T}) = \mathcal{N}(\hat{V}S)$, this null space is determined by the vanishing diagonal entries of the matrix S. It follows that (2.13) holds if and only if there is an index i such that $s_ic_iu_i^Tb \neq 0$. Then $\varphi'(\mu; V) < 0$ and, therefore, $\varphi(\mu; V)$ is strictly decreasing for $\mu > 0$.

Notice that if the assumption (2.13) is not fulfilled, then the derivative $\varphi'(\mu; V)$ vanishes and $\varphi(\mu; V)$ simplifies to the constant function

$$\varphi(\mu; V) = \varphi(V) = b^Tb - \sum_{s_i=0}|u_i^Tb|^2.$$

The second derivative is given by

$$\varphi''(\mu; V) = \sum_{i=1}^{k}\frac{6c_i^4s_i^4}{(\mu c_i^2 + s_i^2)^4}|u_i^Tb|^2 = \sum_{s_i>0}\frac{6\gamma_i^4}{(1+\mu\gamma_i^2)^4}|u_i^Tb|^2 \ge 0. \tag{2.14}$$

Thus, the function $\varphi(\mu; V)$ is strictly convex for $\mu > 0$ under the above assumption on $\mathcal{V}$. We use these properties of $\varphi(\mu; V)$ when designing a zero-finder for the function (2.7); see Subsection 3.2. We remark that the convexity is a result of the formulation (1.3) of the Tikhonov problem instead of the standard formulation with $\lambda = \mu^{-1}$.

Combining the discrepancy principle for updating the regularization parameter and enlarging the search space as described in Subsection 2.1 leads to the following algorithm:

Algorithm 2.1 Generalized Krylov Subspace Tikhonov Regularization Method

Require: Initial basis V_0, $V_0^T V_0 = I$

1: **for** $i = 0, 1, \ldots$ until convergence **do**
2: Find the root μ_i of $f(\mu; V_i) = 0$
3: Solve $(V_i^T T(\mu_i) V_i) y_{\mu_i} = V_i^T A^T b$
4: Compute $r_{\mu_i} = T(\mu_i) V_i y_{\mu_i} - A^T b$
5: Reorthogonalize (optional) $\tilde{r}_{\mu_i} = (I - V_i V_i^T) r_{\mu_i}$
6: Normalize $v_{\text{new}} = \tilde{r}_{\mu_i} / \|\tilde{r}_{\mu_i}\|$
7: Enlarge search space $V_{i+1} = [V_i, v_{\text{new}}]$
8: **end for**
9: Determine approximate Tikhonov solution $x_{\mu_i} = V_i y_{\mu_i}$

Algorithm 2.1 iteratively adjusts the regularization parameter and builds up a search space simultaneously. Generally, "convergence" is achieved already for search spaces of fairly small dimension; see Section 4. Most of the computational work is done in line 4, since determining the zero of $f(\mu)$ in line 2 and solving the projected problem in line 3 is quite inexpensive; see Subsections 3.1 and 3.2.

We can use several convergence criteria in line 1:

- Stagnation of the sequence $\{\mu_i\}$, i.e., the relative change of two consecutive μ_i is small: $|\mu_{i+1} - \mu_i| / \mu_i$ is smaller than a given tolerance.
- The relative change of two consecutive Ritz vectors x_{μ_i} is small, i.e., $\|x_{\mu_{i+1}} - x_{\mu_i}\| / \|x_{\mu_i}\|$ is smaller than a given tolerance.
- The absolute value of the last s elements of the vector y_{μ_i} are several orders of magnitude smaller than the first t elements, i.e., recent increases of the search space do not affect the computed solution significantly.
- The residual r_{μ_i} from line 4 is sufficiently small, i.e., $\|r_{\mu_i}\| / \|A^T b\|$ is smaller than a given tolerance.

We remark that the norm of $Ax_{\mu_i} - b$ cannot be used in a stopping criterion since it might not change much during the course of the computations.

Remark 2.1 *An equivalent formulation of* (1.3) *is obtained by adding a quadratic constraint to the problem* (1.1)*, i.e., we seek to solve*

$$\min_{x \in \mathbb{R}^n} \|Ax - b\|^2 \quad \textit{such that} \quad \|Lx\|^2 = \rho^2.$$

If for this problem an estimate ρ of $\|Lx_{true}\|$ is given (instead of an estimate of $\|e\|$), then we may adapt Algorithm 2.1 so that the constraint is satisfied in each iteration step. This can be be achieved by replacing the function $\varphi(\mu; V)$ in the algorithm by

$$\psi(\mu; V) := \|Lx_\mu\|^2 = \|LV(V^T T(\mu) V)^{-1} V^T A^T b\|^2$$

and determining the zero of the function

$$g(\mu; V) := \psi(\mu; V) - \rho^2 \tag{2.15}$$

in each step of the algorithm. However, monotonicity and convexity of $\psi(\mu; V)$ cannot be guaranteed for all search spaces $\mathcal{V}$. Therefore, the zero of g might not be unique. This makes zero-finding difficult.

3 Computational Considerations

We discuss how to efficiently determine an approximate solution of the large-scale Tikhonov problem (1.3) with Algorithm 2.1. Our aim is to organize the computations so that the overall complexity of the algorithm is $\mathcal{O}(mn)$, i.e., of the same order as the evaluation of a matrix-vector product with a general matrix $A \in \mathbb{R}^{m \times n}$. The different parts of the algorithm are considered in some detail, with the goal of achieving an efficient implementation. Subsection 3.1 focuses on the efficient reuse of information when building up the search space, i.e., on lines 3 – 7 of Algorithm 2.1, and in Subsection 3.2 several zero-finders for updating the regularization parameter in line 2 are developed. Here we either make use of knowledge about the asymptotic behavior of the function $f(\mu; V)$ within an inverse rational interpolation scheme, see [20, 21], or apply the cubically convergent zero-finder described in [26].

3.1 Generating the Search Space

To start Algorithm 2.1, an initial subspace $\mathcal{V}$ is required. A natural choice is a Krylov subspace $\mathcal{K}_\ell(A^TA, A^Tb)$ of low dimension, e.g., of dimension $\ell = 5$. The matrix A^TA (which is $\lim_{\mu\to\infty} T(\mu)$) does not have to be explicitly formed; instead $\ell - 1$ matrix-vector products (MatVecs) are evaluated with the matrix A and ℓ MatVecs with A^T to determine a matrix $V \in \mathbb{R}^{n\times\ell}$, whose columns form an orthonormal basis for $\mathcal{V}$.

For the efficient reuse of information, we store the matrices V, AV, $B_A := A^TAV$, LV, and $B_L := L^TLV$. This requires the evaluation of matrix-vector products with L and L^T. Generally, L is sparse and the computational effort for evaluating the latter MatVecs is much smaller than for the evaluation of MatVecs with A and A^T.

It is convenient to use the QR factorizations of AV and LV in Algorithm 2.1. Let $V \in \mathbb{R}^{n\times k}$ and introduce the factorizations

$$AV = Q_AR_A \quad \text{with} \quad Q_A \in \mathbb{R}^{m\times k},\ R_A \in \mathbb{R}^{k\times k}, \tag{3.1}$$
$$LV = Q_LR_L \quad \text{with} \quad Q_L \in \mathbb{R}^{p\times k},\ R_L \in \mathbb{R}^{k\times k}, \tag{3.2}$$

where the matrices Q_A and Q_L have orthonormal columns, and the matrices R_A and R_L are upper triangular.

We can formulate the computation of the vector $y_\mu = V^T T(\mu)^{-1} V V^T A^T b$ on line 3 of Algorithm 2.1 as a least-squares problem

$$\left\| \begin{bmatrix} AV \\ \mu^{-1/2} LV \end{bmatrix} y_\mu - \begin{bmatrix} b \\ 0 \end{bmatrix} \right\|^2 = \min!$$

and by using the factorizations (3.1) and (3.2), we obtain

$$\left\| \begin{bmatrix} R_A \\ \mu^{-1/2} R_L \end{bmatrix} y_\mu - \begin{bmatrix} Q_A^T b \\ 0 \end{bmatrix} \right\|^2 = \min! \tag{3.3}$$

with the associated normal equations

$$(R_A^T R_A + \mu^{-1} R_L^T R_L) y_\mu = R_A^T d_A, \tag{3.4}$$

where $d_A := d_A(V) := Q_A^T b$. Thus, whenever y_μ has to be computed, we solve the $2k \times k$ least squares problem (3.3).

The residual in line 4 of Algorithm 2.1 can be evaluated by using

$$r_{\mu_i} = T(\mu_i) V y_{\mu_i} - A^T b = B_A y_{\mu_i} + \mu_i^{-1} B_L y_{\mu_i} - A^T b. \tag{3.5}$$

The matrices B_A and B_L are of the same rectangular size, $n \times k$, as V. Therefore, the computational effort required to evaluate MatVecs with B_A and B_L is negligible compared with the effort required to compute a MatVec with the large matrix A.

We note that in exact arithmetic, the residual obtained by evaluating (3.5) gives the same new vector v_{new} for enlarging the search space in line 7 of Algorithm 2.1 as computing

$$r_{\mu_i} = B_A y_{\mu_i} + \mu_i^{-1} B_L y_{\mu_i}, \tag{3.6}$$

since the direction $A^T b$ already is contained in the search space. The latter follows from the fact that $V e_1 = A^T b / \|A^T b\|$, where $e_1 = [1, 0, \ldots, 0]^T$. When computing r_{μ_i} by (3.5) with round-off errors, reorthogonalization in line 5 of Algorithm 2.1 is optional, since by construction of the orthogonal projection, the Galerkin condition $r_{\mu_i} \perp \mathcal{V}$ holds. However, the use of r_{μ_i} from (3.6) requires reorthogonalization in line 5 of Algorithm 2.1. We therefore use (3.5) to compute r_{μ_i}.

When enlarging the search space in line 7 of Algorithm 2.1 by the new vector v_{new}, it is necessary to update the matrices AV, B_A, LV, and B_L. This can be done by two MatVecs involving the matrices A and A^T. Specifically, we compute Av_{new}, $A^T(Av_{\text{new}})$, Lv_{new}, and $L^T(Lv_{\text{new}})$, append a column to the matrices AV, B_A, LV, and B_L, and update the QR factorizations of $AV = Q_A R_A$ and $LV = Q_L R_L$ as follows:

$$A[V, v_{\text{new}}] = [Q_A, \tilde{q}_A] \begin{bmatrix} R_A & r_A \\ 0 & r_a \end{bmatrix}, \tag{3.7}$$

$$L[V, v_{\text{new}}] = [Q_L, \tilde{q}_L] \begin{bmatrix} R_L & r_L \\ 0 & r_l \end{bmatrix}; \tag{3.8}$$

see [8] for a detailed discussion on updating and downdating of QR factorizations. The new vectors are obtained by

$$\begin{aligned} r_A &= Q_A^T(Av_{\text{new}}), \quad q_A = Av_{\text{new}} - r_A, \quad r_a = \|q_A\|, \quad \tilde{q}_A = q_A/r_a, \\ r_L &= Q_L^T(Lv_{\text{new}}), \quad q_L = Lv_{\text{new}} - r_L, \quad r_l = \|q_L\|, \quad \tilde{q}_L = q_L/r_l. \end{aligned}$$

In case of $r_a = 0$, the last expression for $\tilde{q}_A$ is replaced by an arbitrary unit vector such that $Q_A^T \tilde{q}_A = 0$. We proceed analogously when $r_l = 0$. The vector d_A can be updated by $d_A = [d_A; \tilde{q}_A^T b]$. The main computational cost for (3.7) and (3.8) are two matrix-vector products with $Q_A^T \in \mathbb{R}^{k\times m}$ and $Q_L^T \in \mathbb{R}^{k\times p}$. Since k is quite small, the QR factorizations can be updated at negligible cost.

Thus, the computational cost for increasing the dimension of the search space by one is dominated by one MatVec with A and one with A^T. This also is the dominating cost for evaluating one MatVec with the matrix $T(\mu_i)$. The cost for determining the search space $\mathcal{V}$ therefore is acceptable.

3.2 Zero-Finding Methods

This subsection discusses updating strategies for the regularization parameter in line 2 of Algorithm 2.1. The algorithm determines the zero of $f(\mu; V)$ in every iteration. A possible approach is to compute the GSVD of the matrix pair $\{AV, LV\}$ and work with the explicit forms of φ, φ', φ'' given by (2.11), (2.12), (2.14). However, it is expensive to update the GSVD of $\{AV_i, LV_i\}$ in the next step to obtain the GSVD of $\{AV_{i+1}, LV_{i+1}\} = \{A[V_i, v_{\text{new}}], L[V_i, v_{\text{new}}]\}$. We therefore proceed in a different manner.

Since we already have computed the QR factorizations of AV and LV, we can use them to simplify the expression for the function $\varphi(\mu; V)$ and, thereby, also the expression for $f(\mu; V)$. When substituting the factorizations (3.1) and (3.2) into (2.4), we obtain

$$\varphi(\mu; V) = \|AVy_\mu - b\|^2 = b^Tb + (R_Ay_\mu)^T(R_Ay_\mu) - 2(R_Ay_\mu)^Td_A. \tag{3.9}$$

Moreover, the relation $R_A^T d_A = V^T A^T b = \|A^T b\| e_1$ gives

$$\varphi(\mu; V) = b^Tb + (R_Ay_\mu)^T(R_Ay_\mu) - 2\|A^Tb\|e_1^T y_\mu.$$

Thus, we can evaluate $\varphi(\mu; V)$, and therefore also $f(\mu; V) = \varphi(\mu; V) - \delta^2$, by solving a small least-squares problem with a $2k \times k$ matrix consisting of two stacked triangular matrices; cf. (3.3).

When using several values of the monotone function f, not all having the same sign, we can construct bracketing algorithms. In the following, we investigate the asymptotic behavior of $f(\mu; V)$. This will enable us to determine suitable model functions $\tilde{f}$ that approximate f and are used to construct efficient bracketing algorithms. We propose to use a rational inverse interpolation scheme, i.e., $f(\mu^i) \approx 0$ is replaced by $\mu^i = \tilde{f}^{-1}(0)$. Here and in the following μ^i denotes a computed iterate, and μ_k is an approximation of $\bar{\mu}$ associated with the search space $\mathcal{V}$ of dimension k.

Our zero-finders use derivative values of $f(\mu)$ to achieve faster convergence. We investigate the behavior of the first two derivatives of $f(\mu)$ as $\mu \to 0$ and $\mu \to \infty$ in order to determine a suitable model function. The analysis uses the QR factorizations (3.1) and (3.2) of AV and LV.

The derivative of $y_\mu = (R_A^T R_A + \mu^{-1} R_L^T R_L)^{-1} R_A^T d_A$ from (3.4) is given by

$$
\begin{aligned}
y'_\mu \quad &:= \quad \frac{\partial y(\mu)}{\partial \mu} = \mu^{-2}(R_A^T R_A + \mu^{-1} R_L^T R_L)^{-1} R_L^T \\
&\qquad \times R_L (R_A^T R_A + \mu^{-1} R_L^T R_L)^{-1} R_A^T d_A \\
&= \quad \mu^{-2}(R_A^T R_A + \mu^{-1} R_L^T R_L)^{-1} R_L^T R_L y_\mu .
\end{aligned}
\tag{3.10}
$$

The above expression yields

$$
\begin{aligned}
f'(\mu; V) \quad &= \quad 2\left((R_A y_\mu)^T R_A y'_\mu - \mu^{-2}(R_L y_\mu)^T (R_L y_\mu)\right), \\
f''(\mu; V) \quad &= \quad 6 \|R_A y'_\mu\|^2 .
\end{aligned}
\tag{3.11}
$$

Note that $f'(\mu;V) = \varphi'(\mu;V)$ and $f''(\mu;V) = \varphi''(\mu;V)$. The quantity y'_μ can be evaluated similarly as (3.3), i.e., by solving the least-squares problem

$$
\left\| \begin{bmatrix} R_A \\ \mu^{-1/2} R_L \end{bmatrix} y'_\mu - \begin{bmatrix} 0 \\ \mu^{-3/2} R_L y_\mu \end{bmatrix} \right\| = \min! \tag{3.12}
$$

with the associated normal equations

$$
(R_A^T R_A + \mu^{-1} R_L^T R_L) y'_\mu = \mu^{-2} R_L^T R_L y_\mu . \tag{3.13}
$$

We remark that the system matrices in (3.12) and (3.3) are the same. Therefore, we store the QR factorization of $[R_A^T, \mu^{-1/2} R_L^T]^T$ and use it when solving both least-squares problems. This makes the evaluation of the derivative of $f(\mu)$ at $\mu = \mu_j$, when $f(\mu_j)$ already has been computed, cheaper than the initial evaluation of $f(\mu_j)$.

We turn to the asymptotic behavior of f, f', and f''. When investigating the expressions (3.9) and (3.11), the asymptotic behavior of y_μ and y'_μ is important. The limits of the solution y_μ of equation (3.4) are given by

$$
\begin{aligned}
y_\mu^\infty \quad &:= \quad \lim_{\mu \to \infty} y_\mu = \lim_{\mu \to \infty} (R_A^T R_A + \mu^{-1} R_L^T R_L)^{-1} R_A^T d_A \\
&= \quad (R_A^T R_A)^\dagger R_A^T d_A = R_A^\dagger d_A , \\
y_\mu^0 \quad &:= \quad \lim_{\mu \to 0} y_\mu = \lim_{\mu \to 0} (R_A^T R_A + \mu^{-1} R_L^T R_L)^{-1} R_A^T d_A = 0 .
\end{aligned}
$$

Similarly, the limits of the solution y'_μ of (3.13) are

$$\begin{aligned}
(y_\mu^\infty)' &:= \lim_{\mu\to\infty} y'_\mu \\
&= \lim_{\mu\to\infty} (R_A^T R_A + \mu^{-1} R_L^T R_L)^{-1} \mu^{-2} R_L^T R_L y_\mu^\infty = 0, \\
(y_\mu^0)' &:= \lim_{\mu\to 0} y'_\mu \\
&= \lim_{\mu\to 0} (R_A^T R_A + \mu^{-1} R_L^T R_L)^{-1} \mu^{-2} R_L^T \\
&\quad \times R_L (R_A^T R_A + \mu^{-1} R_L^T R_L)^{-1} R_A^T d_A \\
&= (R_L^T R_L)^\dagger R_L^T R_L (R_L^T R_L)^\dagger R_A^T d_A = (R_L^T R_L)^\dagger R_A^T d_A.
\end{aligned}$$

We now are in a position to investigate the function f. Let us begin with the behavior of (3.9) as $\mu \to \infty$, i.e.,

$$\begin{aligned}
\lim_{\mu\to\infty} f(\mu; V) &= \lim_{\mu\to\infty} \|AVy_\mu^\infty - b\|^2 - \delta^2 \\
&= \lim_{\mu\to\infty} (b^T b + (R_A y_\mu^\infty)^T (R_A y_\mu^\infty) - 2(R_A y_\mu^\infty)^T d_A - \delta^2) \\
&= \|b\|^2 + d_A^T (R_A R_A^\dagger) d_A - 2 d_A^T (R_A R_A^\dagger) d_A - \delta^2 \\
&= \|b\|^2 - \|P_{\mathcal{R}(R_A)} d_A\|^2 - \delta^2. \qquad (3.14)
\end{aligned}$$

The limit has to be negative, because otherwise f has no finite zero for this search space. This follows from the fact that $f(\mu; V)$ is monotonically decreasing; cf. (2.12). Therefore, if $\|b\|^2 - d_A^T (R_A R_A^\dagger) d_A > \delta^2$ for the initial matrix V, then Algorithm 2.1 has to enlarge the initial space. The value $\lim_{\mu\to\infty} f(\mu; V)$ decreases monotonically when enlarging the search space. This follows from the structure of $d_A = d_A(V) = Q_A(V)^T b$. We obtain for $V_{\text{new}} = [V, v_{\text{new}}]$ that

$$\begin{aligned}
d_A(V_{\text{new}})^T (R_A R_A^\dagger) d_A(V_{\text{new}}) &= \|P_{\mathcal{R}(R_A)} d_A(V_{\text{new}})\|^2 \\
&= \sum_{R_A(i,i)\neq 0} |Q_A(:,i)^T b|^2 + |\tilde{q}_A^T b|^2. \qquad (3.15)
\end{aligned}$$

Thus, provided that (2.1) holds, it is possible to enlarge the search space until $f(\mu; V)$ has a finite zero. In general, $\|P_{\mathcal{N}(A^T)} b\| \ll \delta$ and $f(\mu; V)$ already has a finite zero for search spaces of small dimension. We remark that if $\lim_{\mu\to\infty} f(\mu; V) < 0$, then this inequality also holds for all subsequent enlarged search spaces. This follows from (3.14) and (3.15).

The behavior of f at the origin is given by

$$\begin{aligned}
\lim_{\mu\to 0} f(\mu; V) &= \|AVy_\mu^0 - b\|^2 - \delta^2 \\
&= \lim_{\mu\to 0} (b^T b + (R_A y_\mu^0)^T (R_A y_\mu^0) - 2(R_A y_\mu^0)^T d_A - \delta^2) \\
&= \|b\|^2 - \delta^2. \qquad (3.16)
\end{aligned}$$

Note that $f(0; V) = f(0) = \|b\|^2 - \delta^2$ is independent of V. We require that $f(0) > 0$, i.e., $\delta > 0$ must be smaller than $\|b\|$. The equations (3.15) and (3.16) are a restriction of the inequalities (2.1) to the search space $\mathcal{V}$: To ensure a finite zero of $f(\mu;V)$, it has to hold

$$\|P_{\mathcal{N}(V^TA^T)}b\| < \delta < \|b\|.$$

Turning to the limits of the first derivative of f, we obtain

$$\begin{aligned}
\lim_{\mu\to\infty} f'(\mu;V) &= \lim_{\mu\to\infty} 2\Big((R_Ay_\mu^\infty)^T R_A(y_\mu^\infty)' - \mu^{-2}(R_Ly_\mu^\infty)^T(R_Ly_\mu^\infty)\Big)\\
&= \lim_{\mu\to\infty} 2\Big(0 - \mu^{-2}(R_LR_A^\dagger d_A)^T(R_LR_A^\dagger d_A)\Big) = 0.
\end{aligned}$$

Thus, the limit is independent of V. Similarly,

$$\begin{aligned}
\lim_{\mu\to 0} \quad f'(\mu;V) &= \lim_{\mu\to 0} 2\left((R_Ay_\mu^0)^T R_A(y_\mu^0)' - \mu^{-2}(R_Ly_\mu^0)^T(R_Ly_\mu^0)\right)\\
&= \lim_{\mu\to 0}\Big(2(R_A(R_A^TR_A+\mu^{-1}R_L^TR_L)^{-1}R_A^Td_A)^T R_A(R_L^TR_L)^\dagger R_A^Td_A\\
&\quad -2\mu^{-2}(R_L(R_A^TR_A+\mu^{-1}R_L^TR_L)^{-1}R_A^Td_A)^T\\
&\quad \times(R_L(R_A^TR_A+\mu^{-1}R_L^TR_L)^{-1}R_A^Td_A)\Big)\\
&= \lim_{\mu\to 0}(-2\|(R_L(\mu R_A^TR_A+R_L^TR_L)^{-1}R_A^Td_A)\|^2\\
&= -2\|(R_L(R_L^TR_L)^\dagger(\|A^Tb\|e_1)\|^2\\
&= -2\|A^Tb\|^2\|(R_L^T)^\dagger e_1\|^2.
\end{aligned} \tag{3.17}$$

While $f(0;V)$ is independent of V, the derivative $f'(0;V)$ is not; the slope decreases monotonically with increasing search space dimension.

Finally, we consider the limits of the second derivative. We obtain

$$\begin{aligned}
\lim_{\mu\to 0} f''(\mu;V) &= 6\|R_A(y_\mu^0)'\|^2\\
&= 6\|R_A(R_L^TR_L)^\dagger R_A^Td_A\|^2\\
&= 6\|A^Tb\|^2\|R_A(R_L^TR_L)^\dagger e_1\|^2
\end{aligned}$$

and

$$\lim_{\mu\to\infty} f''(\mu;V) = 6\|R_A(y_\mu^\infty)'\|^2 = 0.$$

We turn to the design of efficient zero-finders. Newton's method is an obvious candidate. This method works well if a fairly accurate initial approximation of the zero is known. However, if our initial approximation is larger than and not very close to the desired zero, then the first Newton step is likely to give a worse approximation of the zero than the initial approximation; see Figure 1 for a typical plot of $f(\mu)$. The function f is seen to be very steep for small values of $\mu > 0$ and quickly approaches the asymptote as μ increases.

It is interesting to note that the behavior of f close to the zero is not much influenced by the value and the derivative of f at $\mu = 0$. Since the derivative $f'(0;V)$, see (3.17), generally is of very large magnitude, Newtons' method with initial approximation $\mu = 0$ often requires a large number of steps.

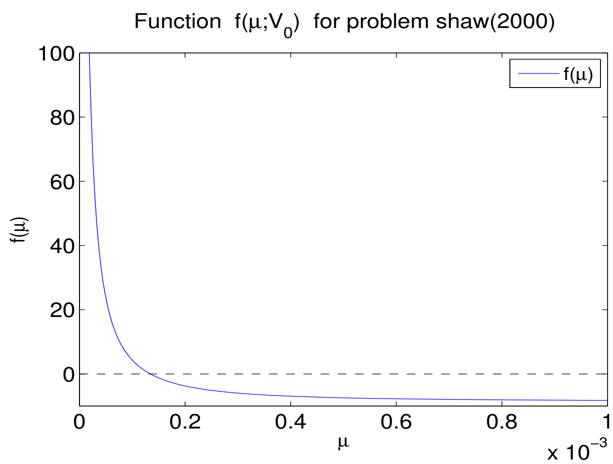

Figure 1: Plot of a typical function $f(\mu)$.

We now describe the construction of a suitable model function for f to be used in a bracketing zero-finder. It is important that the zero-finder accurately models the asymptotic behavior of f as $\mu \to \infty$. We derive a zero-finder based on rational inverse interpolation, which takes this behavior into account. Consider the model function for the inverse of $f(\mu)$,

$$f^{-1} \approx h(f) = \frac{p(f)}{f - f_\infty} \quad \text{with a polynomial} \quad p(f) = \sum_{i=0}^{k-1} a_i f^i, \tag{3.18}$$

where the pole $f_\infty(V) = \|b\|^2 - d_A(V)^T (R_A R_A^\dagger) d_A(V) - \delta^2$ is a function of the columns of V. The degree of the polynomial can be chosen depending on the information of f that is to be used in each step. We propose to use either three function values ($k = 3$) or two function values and two derivative values ($k = 4$). These choices yield small linear systems of equations with a $k \times k$ matrix that have to be solved in each step.

We first consider the use of three pairs $\{\mu^i, f(\mu^i)\}$, $i = 1, 2, 3$, and no derivative information; see also [20]. Assume that the following inequalities are satisfied,

$$\mu^1 < \mu^2 < \mu^3 \quad \text{and} \quad f(\mu^1) > 0 > f(\mu^3); \tag{3.19}$$

otherwise we renumber the μ^i so that (3.19) holds.

If f is strictly monotonically decreasing in $[\mu^1, \mu^3]$, then (3.18) is a rational interpolant of $f^{-1} : [f(\mu^3), f(\mu^1)] \to \mathbb{R}$. Our next iterate is $\mu_{\text{new}} = h(0)$, where the polynomial $p(f)$ is of degree 2. The coefficients a_0, a_1, a_2 are computed by solving the equations $h(f(\mu^i)) = \mu^i$, $i = 1, 2, 3$, which we formulate as a linear system of equations with a 3×3 matrix. In exact arithmetic, $\mu_{\text{new}} \in (\mu^1, \mu^3)$, and we replace μ^1 or μ^3 by μ_{new}, so that the new triplet satisfies (3.19).

Due to round-off errors, the computed value μ_{new} might not be contained in the interval (μ^1, μ^3). In this case we carry out a bisection step, so that the interval is guaranteed to still contain the zero. If we have two positive values $f(\mu^i)$, then we let $\mu^1 = (\mu^2 + \mu^3)/2$; in the case of two negative values $f(\mu^i)$, we let $\mu^3 = (\mu^1 + \mu^2)/2$.

Our second zero-finder uses the pairs $(\mu^i, f(\mu^i))$ and $(\mu^i, f'(\mu^i))$ for $i = 1, 2$. Note that the evaluation of these four pairs is cheaper than the computation of three pairs $(\mu^i, f(\mu^i))$, $1 \leq i \leq 3$, because of the connection between the equations (3.3) and (3.12) for the evaluation of (3.11). The two values μ^i are determined so that the function values $f(\mu^i)$ do not have the same sign, i.e., we require that

$$\mu^1 < \mu^2 \quad \text{and} \quad f(\mu^1) > 0 > f(\mu^2).$$

The next iterate is $\mu_{\text{new}} = h(0)$, where the polynomial $p(f)$ is of degree 3. It is determined by the conditions

$$h(f(\mu^i)) = \mu^i, \qquad h'(f(\mu^i)) = 1/f'(\mu^i), \qquad i = 1, 2,$$

which give rise to a linear system of equations with a 4×4 matrix for the coefficients of the polynomial.

The derivative of the model function is given by

$$h'(f(\mu^i)) = \frac{a_3\left(2f(\mu^i)^3 - 3f_\infty f(\mu^i)^2\right) + a_2\left(f(\mu^i)^2 - 2f_\infty f(\mu^i)\right) - a_1 f_\infty - a_0}{(f(\mu^i) - f_\infty)^2}.$$

Note that $1/f'(\mu^i)$ is the derivative of f^{-1} at $f(\mu^i)$. The value μ_{new} replaces the value μ^i on the same side of the root. In case $\mu_{\text{new}} \notin (\mu^1, \mu^2)$ (e.g. due to round-off errors), a bisection step is carried out. Depending on the sign of $f((\mu^1 + \mu^2)/2)$, the appropriate μ^i is updated.

When the zero of f is approached, the condition number of the 3×3 or 4×4 matrices become very large. We therefore represent $p(f)$ in terms of a basis of Chebyshev polynomials for the intervals $[f(\mu^3), f(\mu^1)]$ or $[f(\mu^2), f(\mu^1)]$, respectively.

An alternative approach is the special zero-finder for Tikhonov problems proposed in [26], originally developed for problems in standard form. The conditions for cubic convergence are fulfilled, i.e., the function $f(\mu; V) : \mathbb{R}_+ \to \mathbb{R}$ is three times continuously differentiable and satisfies

$$f'(\mu; V) < 0, \quad f''(\mu; V) > 0, \quad \mu > 0,$$

as well as

$$\lim_{\mu \to 0} f(\mu; V) > 0, \quad \lim_{\mu \to \infty} f(\mu; V) < 0.$$

It follows from (2.12) and (2.14) that $f(\mu; V)$ is decreasing and convex. This zero-finder is compared to the bracketing algorithms and Newton's method in the following section.

If the requirement (2.13) on the search space $\mathcal{V}$ is violated, then it is not possible to find a zero of $f(\mu; V)$ (except when $f(\mu; V) \equiv 0$). This situation would arise, if we choose the initial search space $\mathcal{V}$ to be a subset of $\mathcal{N}(L)$. Then the matrix LV is the zero-matrix and $s_1 = \ldots = s_k = 0$ in the GSVD (2.9)-(2.10). Such a choice of $\mathcal{V}$ is fairly natural. For example, we may know that the desired solution x_{true} is approximately a multiple of the vector $w = [1, 1, \ldots, 1]^T$. We then would like $w \in \mathcal{V}$. Moreover, we may want to use a regularization matrix L, such that $w \in \mathcal{N}(L)$, because vectors in $\mathcal{N}(L)$ are not damped by Tikhonov regularization (1.3); see, e.g., [27, Example 4.2] for an illustration. A simple way to handle this situation is to increase the dimension of the initial search space $\mathcal{V}$, e.g., by carrying out a few more Lanczos steps, until (2.13) is satisfied.

4 Numerical Examples

We use two test examples from Hansen's *Regularization Tools* [17] to illustrate the performance of Algorithm 2.1. The matrices in both examples are of ill-determined rank and numerically singular.

The MATLAB functions `heat` and `blur` yield square matrices $\hat{A}$, the right-hand sides $\hat{b}$, and the solutions x_{true}, with $\hat{A}x_{\text{true}} = \hat{b}$. Adding white Gaussian noise e to b_{true} yields the error-contaminated vector b in (1.1); cf. (1.2). We refer to the quotient

$$\sigma := \frac{\|e\|}{\|b_{\text{true}}\|}$$

as the *noise level*. The computations are carried out on a PentiumR4 computer with 3.4 GHz and 8GB RAM under MATLAB R2008a.

Example 4.1. In this example, we solve an overdetermined linear system of equations by stacking the matrix and right-hand side of the inverse heat equation `heat` (κ=5), i.e.,

$$A = \begin{bmatrix} \hat{A} \\ \hat{A} \end{bmatrix}, \qquad b_{\text{true}} = \begin{bmatrix} \hat{b} \\ \hat{b} \end{bmatrix},$$

with the matrix $A \in \mathbb{R}^{400 \times 200}$ and the error-contaminated vector b defined as described above with noise level $\sigma = 1 \cdot 10^{-2}$. Stacked problems of this kind arise when two measurements b with different errors are available. The condition number $\kappa(A) := \|A\|\|A^\dagger\|$ of A is larger than 10^{17}. Thus, the matrix is numerically singular. The parameter η in (1.8) is set to 1.1. The regularization matrix L is chosen to be a scaled discrete first order derivative operator in one space-dimension,

$$L_1 = \begin{bmatrix} 1 & -1 & & \\ & \ddots & \ddots & \\ & & 1 & -1 \end{bmatrix} \in \mathbb{R}^{(n-1) \times n} \tag{4.1}$$

with $n = 200$. This regularization matrix has a structure that allows transformation of (1.3) to standard form before solution. We will not exploit this, but instead illustrate different aspects of the performance of Algorithm 2.1.

The initial search space is the Krylov subspace $\mathrm{span}\{V_0\} = \mathcal{K}_\ell(A^TA, A^Tb)$ for $\ell = 7$. When $\ell < 7$, the condition $\lim_{\mu\to\infty} f(\mu, V_0) < 0$ does not hold; cf. equation (3.14). Note that the index $\ell = 7$ is not known a priori but can be determined on-the-fly when building the initial Krylov subspace by cheaply evaluating (3.14) with the help of (3.15) for each value of ℓ. We would like to illustrate the convergence of several interesting quantities and therefore do not terminate the computations with Algorithm 2.1 until $\dim(\mathcal{V}) = 200$.

Figure 2(a) displays the sequence $\{\mu_k\}$, and we observe that the value $\bar{\mu} \approx 8.07$ is approached very quickly as $\dim(\mathcal{V})$ increases. We recall that $\bar{\mu}$ is the solution of $\|Ax_{\bar{\mu}} - b\|^2 - \delta^2 = 0$, i.e., of $f(\mu, V) = 0$ when $\dim(\mathcal{V}) = 200$. The rate of convergence of the regularization parameter as a function of the dimension k of the search space $\mathcal{V}$ is illustrated by Figure 2(b); the figure shows the relative change of the regularization parameter, $|\mu_{k+1} - \mu_k|/\mu_k$, as a function of $k = \dim(\mathcal{V})$. After 30 iterations, the relative change drops to zero (and therefore cannot be displayed with a semilogarithmic plot). Other quantities on which the stopping criterion for Algorithm 2.1 may be based are shown in Figures 2(c)-(e). The relative change of two consecutive approximate solutions of (1.1), $\|x_{\mu_{k+1}} - x_{\mu_k}\|/\|x_{\mu_k}\|$ for $k = 0, 1, 2, \ldots$, is shown in Figure 2(c) and the corresponding relative residual norms, $\|r(x_{\mu_k})\|/\|A^Tb\|$, $k = 0, 1, 2, \ldots$, are displayed in Figure 2(d); cf. line 4 of Algorithm 2.1.

The magnitude of the entries of the vector $y_{\mu_{200}}$ (cf. equation (2.5)) for $\mathcal{V} = \mathbb{R}^{200}$ are depicted in Figure 2(e), and the exact solution x_{true} together with the computed approximation $x_{\mu_{40}}$ are shown in Figure 2(f). The relative error of the computed approximate solution is $\|x_{\mu_{40}} - x_{\mathrm{true}}\|/\|x_{\mathrm{true}}\| = 1.85 \cdot 10^{-2}$. All four quantities displayed in Figures 2(b)-(e), or some combination of them, can be used as stopping criteria for Algorithm 2.1.

Notice that the convergence of the regularization parameter is not monotonic; see Figure 2(a). It is clearly visible that the sequence $\{\mu_k\}$ oscillates around $\bar{\mu}$ when the dimension k of $\mathcal{V}$ increases and is small. The values μ_k oscillate around $\bar{\mu}$ also when k is large, but this is not visible in the figure. Non-monotonic behavior also can be observed for the other quantities plotted in Figures 2(c)-(e).

We compare four zero-finding algorithms for the subproblems $f(\mu, V_k) = 0$ in line 2 of Algorithm 2.1. Each time the search space is enlarged, the zero-finder comes into play. We apply the two bracketing algorithms described above, i.e., the *3-point* zero-finder using three pairs $(\mu^i, f(\mu^i, V_k))$, $i = 1, 2, 3$, or the *4-point* zero-finder using two function-values and two derivative-values of $f(\mu, V_k)$. The third zero-finder is Newton's method and the last one is the *Cubic*ally convergent method from [26].

The initial value for the regularization parameter is chosen to be $\mu_7^0 = 0.01$ and the dimension of the initial search space is 7. When, subsequently, the dimension of the search space is increased, the initial value of the regularization parameter is set to be equal to the last calculated value, i.e., we set $\mu_{k+1}^0 = \mu_k$.

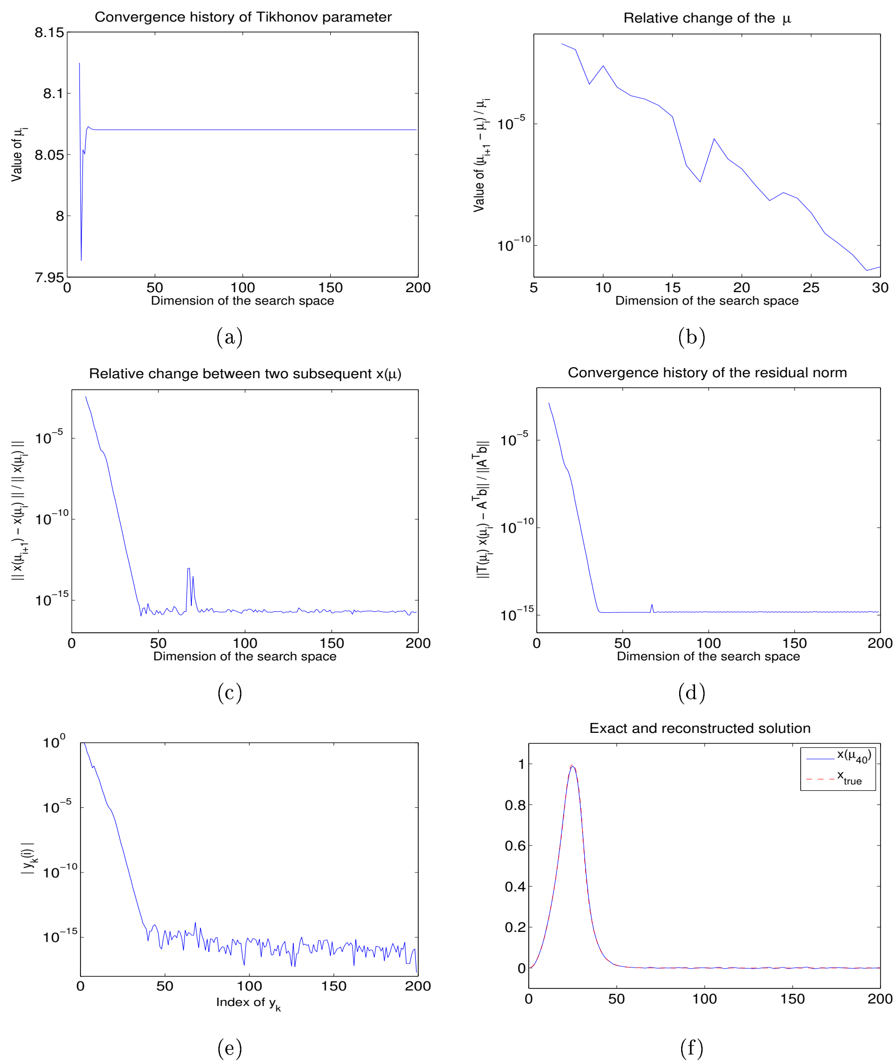

Figure 2: Convergence histories for Example 4.1.

We start the bracketing zero-finders by first determining values μ^i, such that not all $f(\mu^i)$ are of the same sign. Such values can be determined by multiplying available values of the regularization parameter by 0.01 or 100 depending on the sign of $f(\mu, V)$. After very few steps this gives an interval that contains the root of $f(\mu, V)$.

Table 1: Number of iterations of the zero-finders for Example 4.1.

zero-finder	1st iter.	2nd iter.	3rd iter.	ith iter.
3-point	>20	3	3	$1-2$
4-point	7	2	1	$1-2$
Newton	16	4	3	$1-3$
Cubic	7	3	3	$1-2$

Table 1 shows the number of iterations required by the zero-finders considered. After three subspace enlargements, the values μ_k do not change much and only few iterations with the zero-finders are required. The first entry of the second column of Table 1 indicates that more than 20 (inner) iterations are needed before the stopping criterion for the zero-finder is satisfied. In this situation, we terminate the iterations with the zero-finder at step 20 and use the best available approximation of the zero. This is the only time when a zero-finder did not satisfy the stopping criterion. For the non-bracketing zero-finders, we use the stopping criterion

$$\frac{|f(\mu_k^i)|}{\delta^2} < 1 \cdot 10^{-8},$$

where μ_k^i is the current approximation of the desired zero. We terminate the iterations with the bracketing zero-finders when

$$\frac{\min\{|f(\mu_k^1)|, |f(\mu_k^2)|\}}{\delta^2} < 1 \cdot 10^{-8},$$

where μ_k^1 and μ_k^2 are best bracketing approximations of the zero. The ith iteration in the fifth column shows the number of iterations with the zero-finder required after all subsequent 190 enlargements of the search space. For this example all methods carried out between one and three steps.

All zero-finders produce approximations of x_{true} of similar quality and solve the problems $f(\mu, V_k)$, $k = 7, \ldots, 200$, to desired accuracy (except for the 3-point bracketing algorithm for the first search space enlargement). Indeed, for the final computed approximation of x_{true}, $x_{\mu_{200}}$, we have

$$\frac{|\|Ax_{\mu_{200}} - b\|^2 - \delta^2|}{\delta^2} < 7 \cdot 10^{-12}$$

for all zero-finders. This value is much smaller than the desired accuracy of $1 \cdot 10^{-8}$. This is possibly due to the fact that at least one step of the zero-finder is carried out after each search space expansion.

Table 1 shows the 4-point bracketing algorithm to be slightly superior to the cubic solver. The 3-point bracketing and Newton's methods are clearly inferior. For the initial search space, Newton's method and the 3-point zero-finder require more than twice as many iterations as the other two zero-finders. The cost of one iteration is about the same for each one of the four zero-finders. The main effort is the factorization of the $2k \times k$ matrix $[R_A^T, \mu^{-1/2} R_L^T]^T$; cf. equations (3.3) and (3.12).

Example 4.2. We consider the restoration of a greyscale image that is represented by an array of 100×100 pixels. The pixels are stored columnwise in a vector in $\mathbb{R}^{10000}$. The vector x_{true} represents the uncontaminated image. A block Toeplitz with Toeplitz blocks blurring matrix $A \in \mathbb{R}^{10000\times 10000}$ is determined with the function `blur` from [17] using the parameter values $band =$ 5 (which is the half-bandwidth of each 100×100 Toeplitz block) and $sigma = 1.0$ (which determines the width of the underlying Gaussian point spread function). The matrix A has 7.7×10^5 nonzero entries. We add white Gaussian noise corresponding to the noise level $\sigma = 1 \cdot 10^{-2}$. This determines the vector b in (1.1). The vector x_{true} is assumed not to be available. We would like to determine an accurate restoration of x_{true} given A and b. The factor η in (1.8) is set to 1.05.

Different regularization matrices L and zero-finders are compared. We use the first-order discrete derivative operator for two space-dimensions

$$L_{1,2D} = \begin{bmatrix} L_1 \otimes I_n \\ I_n \otimes L_1 \end{bmatrix}$$

with L_1 defined by (4.1) with $n = 100$ and I_n the identity matrix of order 100. The second order discrete derivative operator in two space-dimensions

$$L_{2,2D} = \begin{bmatrix} L_2 \otimes I_n \\ I_n \otimes L_2 \end{bmatrix}$$

also is considered, where

$$L_2 = \begin{bmatrix} -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \end{bmatrix} \in \mathbb{R}^{(n-2)\times n}, \qquad n = 100.$$

The A-weighted pseudoinverses of $L_{1,2D}$ and $L_{2,2D}$ are unattractive to use. We compare the performance of Algorithm 2.1 for the regularization matrices $L_{1,2D}$, $L_{2,2D}$, and $L = I$. For the latter regularization matrix, the generalized Krylov subspaces $\mathcal{V}$ determined by Algorithm 2.1 reduce to the standard Krylov subspaces $\mathcal{K}_k(A^TA, A^Tb)$ and our iterative method simplifies to LSQR.

The initial search space is chosen to be $\text{span}\{V_0\} = \mathcal{K}_{10}(A^TA, A^Tb)$. The stopping criterion in line 1 of Algorithm 2.1 is to proceed until the dimension of the search space reaches 40. This choice is rather arbitrary and can be replaced by one or several of the convergence criteria described above. The convergence histories of the most interesting quantities when using the regularization matrix $L = L_{1,2D}$ is shown in Figure 3. The graphs are similar for the regularization matrices $L = L_{2,2D}$ and $L = I$. The latter graphs therefore are not shown.

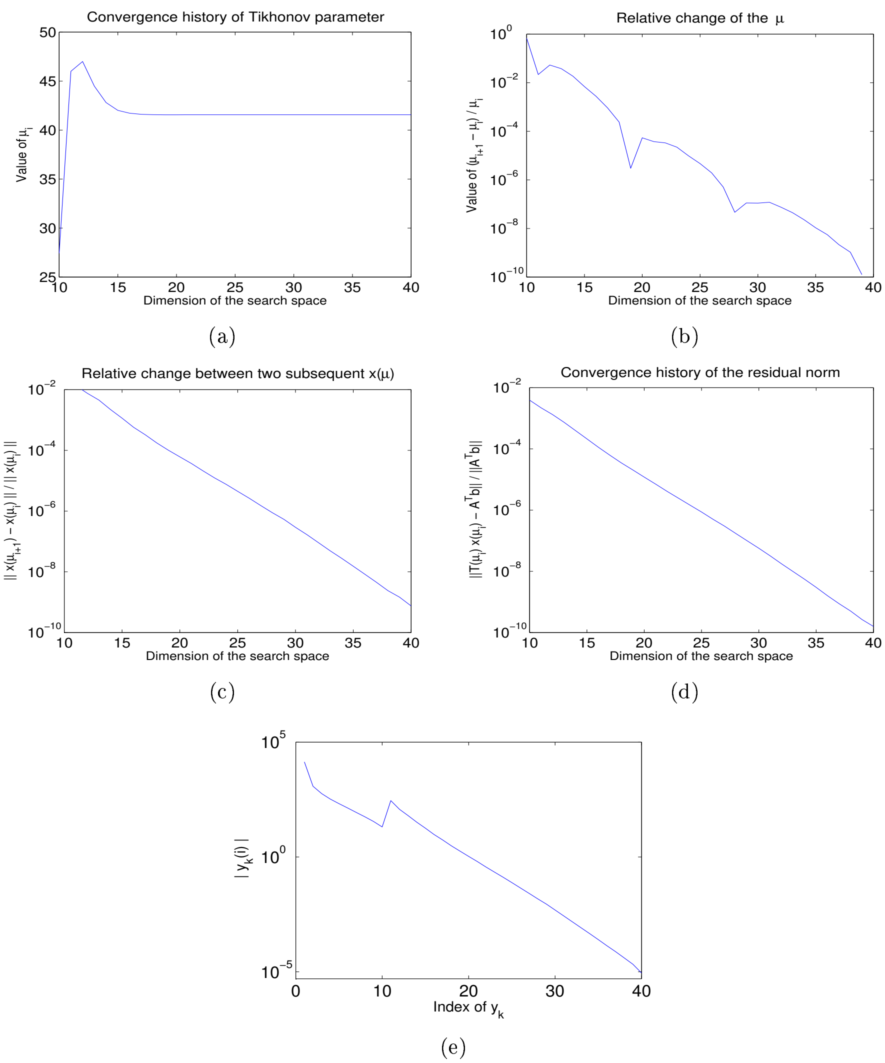

Figure 3: Convergence histories for the restoration of *Danny* using the regularization matrix $L_{1,2D}$.

Figure 3(a) displays the sequence $\{\mu_k\}$. Similarly as in Example 4.1, convergence is not monotonic and the regularization parameter stagnates quite quickly; see also Figure 3(b). Other quantities on which a stopping criterion for Algorithm 2.1 can be based are displayed in Figures 3(c)-(e). The relative change of two consecutive approximations x_{μ_k} and the corresponding relative residual norm $\|r(x_{\mu_k})\|/\|A^T b\|$ are shown in Figure 3(c) and (d), respectively. Both quantities decrease by 7 orders of magnitude. The magnitude of the entries of the vector $y_{\mu_{40}}$ are displayed in Figure 3(e); they decrease by 9 orders of magnitude, but not monotonically.

Figure 4 shows the original (blur- and noise-free) image, the blurred and noisy image, and several restorations. The first row of Figure 4 depicts the original image as well as the blur- and noise-contaminated image. The first restored image in the second row is obtained by using the discrete Laplace operator $L_{2,2D}$ as regularization matrix and applying 30 outer iterations in Algorithm 2.1. Since the initial search space is of dimension 10, the search space is of dimension $k = \dim(\mathcal{V}) = 40$ at termination. The relative error in the computed restoration is $\|x_{\mu_{40}} - x_{\text{true}}\|/\|x_{\text{true}}\| = 3.97 \cdot 10^{-2}$.

The restoration shown by the second image in row two is for $L = I$. The termination criterion is the same as above. The computed restoration has relative error $4.28 \cdot 10^{-2}$. The last row displays two restored images obtained with $L = L_{1,2D}$; the first one corresponds to $\dim(\mathcal{V}) = 20$ and the second one to $\dim(\mathcal{V}) = 40$. The difference between these images is very small. Both restorations have a relative error of $3.86 \cdot 10^{-2}$. The relative error of the blurred and noisy image is $\|b - x_{\text{true}}\|/\|x_{\text{true}}\| = 8.20 \cdot 10^{-2}$.

Figure 4 shows the regularization matrix $L = I$ to give the worst restoration; the restored image can be seen to contain a lot of "freckles". Moreover, the relative error is larger than for the other regularization matrices. The quality of the restorations obtained with $L_{1,2D}$ and $L_{2,2D}$ is about the same. We find the images in the bottom row obtained with $L_{1,2D}$ to be slightly sharper than the image determined with $L_{2,2D}$. Also the relative error is slightly smaller. Since the two restored images in the bottom row of Figure 4 are nearly indistinguishable, we conclude that it suffices to use the low-dimensional search space $\mathcal{V}$ of dimension 20.

We compared the four zero-finders for the regularization matrix $L = L_{1,2D}$. The starting value for the Tikhonov parameter is chosen to be $\mu_{10}^0 = 1$, where the subscript 10 shows the dimension of the initial search space. The computations are similar to those of Example 4.1. Table 2 shows the number of iterations required by each zero-finder.

Table 2: Number of iterations of the zero-finders for Example 4.2.

zero-finder	1st iter.	2nd iter.	3rd iter.	ith iter.
3-point	>20	11	3	$2-3$
4-point	4	3	2	1
Newton	11	6	4	$1-4$
Cubic	7	4	3	$1-3$

The convergence behavior illustrated by Table 2 is analogous to the behavior

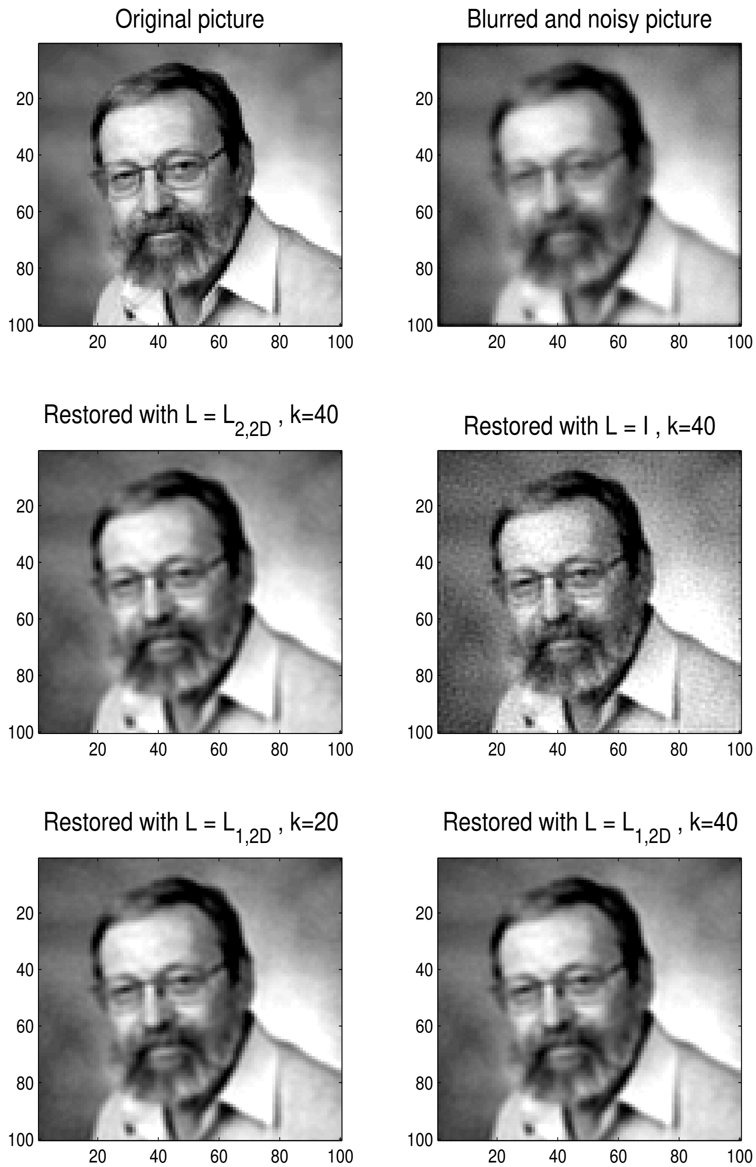

Figure 4: Original, blurred, and restored *Danny*.

displayed by Table 1. For all zero-finders the computed restorations $x_{\mu_{40}}$ satisfy

$$\frac{|\|Ax_{\mu_{40}} - b\|^2 - \delta^2|}{\delta^2} < 4 \cdot 10^{-11}.$$

This accuracy is much higher than the required accuracy of $1 \cdot 10^{-8}$.

Remark 4.1 *The bracketing zero-finders require an initial value of μ that is larger than $\bar{\mu}$. This may result in underregularization. However, the fact that we determine an approximate solution in a search space $\mathcal{V}$ of dimension much smaller than the problem dimension has a regularizing effect. We have not observed difficulties with the bracketing zero-finders due to underregularization.*

Remark 4.2 *We observed in the examples that the projected matrices $V^T T(\mu_k) V$ are "essentially" tridiagonal, i.e., the off-tridiagonal elements are small compared to elements in the tridiagonal part of the matrices. The relative size of the entries depends on the problem and on the changes of the regularization parameter during the execution of the algorithm. This phenomenon may be due to the fast convergence of the regularization parameter, which gives fast convergence of the matrices $\{T(\mu_k)\}$ as well.*

5 Conclusions

A new iterative method for Tikhonov regularization problems with general regularization matrices is presented and zero-finders for determining the regularization parameter are derived. Computed examples suggest several stopping criteria and indicate that search spaces of fairly small dimension suffice.

References

[1] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716 – 723, 1974.

[2] Å. Björck. A bidiagonalization algorithm for solving large and sparse ill-posed systems of linear equations. *BIT*, 28:659 – 670, 1988.

[3] D. Calvetti, G. H. Golub, and L. Reichel. Estimation of the L-curve via Lanczos bidiagonalization. *BIT*, 39:603 – 619, 1999.

[4] D. Calvetti, P. C. Hansen, and L. Reichel. L-curve curvature bounds via Lanczos bidiagonalization. *Electron. Trans. Numer. Anal.*, 14:20 – 35, 2002.

[5] D. Calvetti, S. Morigi, L. Reichel, and F. Sgallari. Tikhonov regularization and the L-curve for large, discrete ill-posed problems. *J. Comput. Appl. Math.*, 123:423 – 446, 2000.

[6] D. Calvetti and L. Reichel. Tikhonov regularization of large linear problems. *BIT*, 43:263 – 283, 2003.

[7] J. Chung, J. G. Nagy, and D. P. O'Leary. A weighted-GCV method for Lanczos-hybrid regularization. *Electron. Trans. Numer. Anal.*, 28:149 – 167, 2008.

[8] J. W. Daniel, W. B. Gragg, L. Kaufman, and G. W. Stewart. Reorthogonalization and stable algorithms for updating the Gram-Schmidt QR factorization. *Math. Comp.*, 30:772 – 795, 1976.

[9] L. Eldén. A weighted pseudoinverse, generalized singular values, and constrained least squares problems. *BIT*, 22:487 – 502, 1982.

[10] H. W. Engl, M. Hanke, and A. Neubauer. *Regularization of Inverse Problems.* Kluwer, Dordrecht, The Netherlands, 1996.

[11] G. H. Golub, M. Heath, and G. Wahba. Generalized cross validation as a method for choosing a good ridge parameter. *Technometrics*, 21:215 – 223, 1979.

[12] G. H. Golub and C. F. Van Loan. *Matrix Computations.* The John Hopkins University Press, Baltimore and London, 3rd edition, 1996.

[13] G.H. Golub and U. von Matt. Tikhonov regularization for large scale problems. In G.H. Golub, S.H. Lui, F.T. Luk, and R.J. Plemmons, editors, *Scientific Computing*, pages 3 – 26, Berlin, 1997. Springer.

[14] M. Hanke. On Lanczos based methods for the regularization of discrete ill-posed problems. *BIT*, 41:1008 – 1018, 2001.

[15] P. C. Hansen. Analysis of discrete ill-posed problems by means of the L-curve. *SIAM Review*, 34(4):561 – 580, 1992.

[16] P. C. Hansen. *Rank-Deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion.* SIAM, Philadelphia, 1998.

[17] P. C. Hansen. Regularization tools version 4.0 for Matlab 7.3. *Numer. Algorithms*, 46:189 – 194, 2007.

[18] M. E. Hochstenbach and L. Reichel. An iterative method for Tikhonov regularization with a general linear regularization operator. *J. Integral Equations Appl.*, 22:463 – 480, 2010.

[19] M. E. Kilmer, P. C. Hansen, and M. I. Español. A projection-based approach to general-form Tikhonov regularization. *SIAM J. Sci. Comput.*, 29(1):315 – 330, 2007.

[20] J. Lampe and H. Voss. A fast algorithm for solving regularized total least squares problems. *Electron. Trans. Numer. Anal.*, 31:12 – 24, 2008.

[21] J. Lampe and H. Voss. Solving regularized total least squares problems based on eigenproblems. *Taiwanese J. Math.*, 14:885 – 909, 2010.

[22] C. L. Lawson and R. J. Hanson. *Solving Least Squares Problems*. Prentice–Hall, Englewood Cliffs, NJ, 1974.

[23] V. A. Morozov. On the solution of functional equations by the method of regularization. *Soviet. Math. Dokl.*, 7:414 – 417, 1966.

[24] D. P. O'Leary and J. A. Simmons. A bidiagonalization-regularization procedure for large scale discretizations of ill-posed problems. *SIAM J. Sci. Stat. Comput.*, 2(4):474 – 489, 1981.

[25] L. Reichel, F. Sgallari, and Q. Ye. Tikhonov regularization based on generalized Krylov subspace methods. Submitted for publication.

[26] L. Reichel and A. Shyshkov. A new zero-finder for Tikhonov regularization. *BIT Numerical Mathematics*, 48:627 – 643, 2008.

[27] L. Reichel and Q. Ye. Simple square smoothing regularization operators. *Electron. Trans. Numer. Anal.*, 33:63 – 83, 2009.

[28] G. Schwarz. Estimating the dimension of a model. *Annals of Statistics*, 6(2):461 – 464, 1978.