# On the choice of solution subspace for nonstationary iterated Tikhonov regularization

Guangxin Huang * Lothar Reichel † Feng Yin ‡

*Dedicated to Gerhard Opfer on the occasion of his 80th birthday.*

**Abstract**

Tikhonov regularization is a popular method for the solution of linear discrete ill-posed problems with error-contaminated data. Nonstationary iterated Tikhonov regularization is known to be able to determine approximate solutions of higher quality than standard Tikhonov regularization. We investigate the choice of solution subspace in iterative methods for nonstationary iterated Tikhonov regularization of large-scale problems. Generalized Krylov subspaces are compared with Krylov subspaces that are generated by Golub–Kahan bidiagonalization and the Arnoldi process. Numerical examples illustrate the effectiveness of the methods.

**Keywords:** ill-posed problem, nonstationary iterated Tikhonov regularization, Golub–Kahan bidiagonalization, Arnoldi process, Krylov subspace, generalized Krylov subspace

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

## 1 Introduction

We would like to compute an approximate solution of large-scale least-squares problems of the form

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

with a matrix $A \in \mathbb{R}^{m\times n}$, whose singular values gradually decay to zero without a significant gap. In particular, $A$ is severely ill-conditioned and possibly singular. Minimization problems (1.1) with a matrix of this kind are commonly

*Geomathematics Key Laboratory of Sichuan, College of Management Science, Chengdu University of Technology, Chengdu, 610059, P. R. China. E-mail: huangx@cdut.edu.cn

†Department of Mathematical Sciences, Kent State University, Kent, OH 44242, USA. E-mail: reichel@math.kent.edu

‡Geomathematics Key Laboratory of Sichuan, Chengdu University of Technology, Chengdu, 610059, P. R. China; School of Sciences, Sichuan University of Science and Engineering, Zigong, 643000, P. R. China. E-mail: fyin@suse.edu.cn

referred to as discrete ill-posed problems. They arise, for instance, from the discretization of ill-posed problems such as Fredholm integral equations of the first kind; see, e.g., [12, 15]. For ease of notation, we will assume that $m \geq n$, however, the methods described easily can be adapted to the case $m < n$.

The vector $b \in \mathbb{R}^m$ in linear discrete ill-posed problems that arise in applications represents measured data and is contaminated by an error $e \in \mathbb{R}^m$, that stems from measurement inaccuracies and possibly also from discretization. Let $b_{\text{true}} \in \mathbb{R}^m$ denote the unknown error-free vector associated with the vector $b$ in (1.1). Then $b$ can be written as

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

We will assume that a bound $\delta$ for the norm of the error in (1.2) is known, i.e.,

$$\|e\| \leq \delta, \tag{1.3}$$

and that the linear system of equations

$$Ax = b_{\text{true}} \tag{1.4}$$

associated with the least-squares problems (1.1) is consistent. We would like to determine an approximation of the solution of minimal Euclidean norm, $x_{\text{true}}$, to (1.4) by computing a suitable approximate solution of (1.1). Throughout this paper $\|\cdot\|$ denotes the Euclidean vector norm or spectral matrix norm.

Because of the ill-conditioning of $A$ and the error $e$ in $b$, straightforward solution of (1.1) typically does not yield a meaningful approximation of $x_{\text{true}}$. Therefore, one often replaces (1.1) by a nearby problem that is less sensitive to the error $e$. This replacement is referred to as regularization. The possibly most popular regularization method is due to Tikhonov. This method replaces (1.1) by a penalized least-squares problem of the form

$$\min_{x \in \mathbb{R}^n} \left\{ \|Ax - b\|^2 + \mu^{-1} \|Lx\|^2 \right\}, \tag{1.5}$$

where $L \in \mathbb{R}^{p \times n}$ is called the regularization matrix and the scalar $\mu > 0$ the regularization parameter. When $L$ is the identity matrix, the Tikhonov minimization problem (1.5) is said to be in *standard form*, otherwise it is said to be in *general form*. The use of the regularization parameter $\mu$ instead of $1/\mu$ is commented on below. We assume that

$$\mathcal{N}(A) \cap \mathcal{N}(L) = \{0\}, \tag{1.6}$$

where $\mathcal{N}(M)$ denotes the null space of the matrix $M$. Then the Tikhonov minimization problem (1.5) has a unique solution $x_\mu$ for any $\mu > 0$. The value of $\mu$ determines how sensitive $x_\mu$ is to the error $e$ and how close $x_\mu$ is to $x_{\text{true}}$; see, e.g., Engl et al. [12] for discussions on Tikhonov regularization.

We would like to determine a suitable value of the regularization parameter $\mu > 0$ and an approximation of the associated solution $x_\mu$ of (1.5) by a

projection-based nonstationary iterated Tikhonov regularization method that can be applied to large-scale problems. Nonstationary iterated Tikhonov regularization replaces the solution of (1.5) by the solution of a sequence of minimization problems

$$\min_{x\in\mathbb{R}^n}\{\|Ax-b\|^2+\mu_k^{-1}\|L(x-x_{k-1})\|^2\},\qquad k=1,2,\dots\ , \tag{1.7}$$

with solutions $x_k$, $k=1,2,\dots$ . Here $x_0$ is an initial approximate solution. Our interest in nonstationary iterated Tikhonov regularization stems from the fact that the approximations $x_k$ of $x_{\rm true}$ computed by this method for $L=I$ for a suitable choice of regularization parameters $\mu_k$ can give more accurate approximations of $x_{\rm true}$ than approximations determined by Tikhonov regularization in standard form; see [11, 12, 14]. Recent successful applications of modifications of nonstationary iterated Tikhonov regularization with $L=I$ are reported in [1, 7].

Small to medium-sized minimization problems (1.7) can be solved conveniently by first computing the generalized singular value decomposition (GSVD) or related decompositions of the matrix pair $\{A,L\}$; see [8, 15] for details. In this paper, we are concerned with the situation when the matrices $A$ and $L$ are too large to make it feasible to compute these decompositions.

Approximations of the solution $x_\mu$ of large-scale Tikhonov regularization problems in standard form, i.e., of (1.5) with $L=I$, can be computed by carrying out a suitable (small) number of steps of Golub–Kahan bidiagonalization of $A$ and then solving the small reduced Tikhonov minimization problem so obtained; see, e.g., [2, 4, 25] and references therein for several solution methods based on this approach. The computed approximation $x_k$, for $k\geq 1$, of $x_{\rm true}$ determined in this manner lives in the Krylov subspace

$$\mathcal{K}_k(A^TA,A^Tb)=\mathrm{span}\{A^Tb,(A^TA)A^Tb,\dots,(A^TA)^{k-1}A^Tb\}, \tag{1.8}$$

where the superscript $^T$ denotes transposition. These solution subspaces also have been applied to the solution of large-scale Tikhonov minimization problems (1.5) in general form; see [17]. We are interested in developing solution methods for the minimization problems (1.7) that are based on partial Golub–Kahan bidiagonalization of $A$ and determine approximate solutions in the subspaces (1.8).

Tikhonov regularization problems in standard form with a large square matrix $A$ also can be solved by first reducing $A$ to a small upper Hessenberg matrix by applying a few steps of the Arnoldi process to $A$, and then solving the reduced problem. This approach, and some variations thereof, are discussed in [5, 9, 24]. The computed approximate solution $x_k$ lives in a Krylov subspace of the form

$$\mathcal{K}_k(A,A^qb)=\mathrm{span}\{A^qb,A^{q+1}b,\dots,A^{q+k-1}b\} \tag{1.9}$$

for some $k\geq 1$ and a fixed small integer $q\geq 0$. A nice recent review of some of these methods is presented by Gazzola et al. [13]. The solution subspaces (1.9) also can be applied when $L\neq I$; computed examples are presented in

[13] for $q = 0$. We will develop solution methods for (1.7) that determine approximate solutions in subspaces of the form (1.9). Our interest in using these solution subspaces instead of spaces of the form (1.8) stems from the property that solution methods that use the former typically require fewer matrix-vector product evaluations than solution methods based on the latter.

Recently, we developed an iterative method for the solution of (1.7) in which approximate solutions were determined in generalized Krylov subspaces [18]. The solution subspaces in this method are expanded in a manner first proposed by Voss [31]. This approach has recently been applied in several methods for the solution of large-scale linear discrete ill-posed problems; see [20, 22].

It is the purpose of the present paper to compare solution methods for (1.7) based on the Krylov subspaces (1.8) and (1.9), as well as on the generalized Krylov subspaces applied in [18]. These methods have been chosen in our comparison because they perform well, are simple to implement, and can be applied to a variety of regularization matrices. When the regularization matrix $L$ has a special structure, such as being banded with a small bandwidth, or is an orthogonal projection, the Tikhonov minimization problem (1.5) can be transformed to standard form in a fairly inexpensive manner; see Eldén [10], Hansen [15], or Morigi et al. [23] for details. After this transformation the nonstationary iterated Tikhonov minimization problems (1.7) can be solved by available techniques for problems in standard form. Also reduction methods described in [19, 27, 28] and in references therein can be applied to the solution of (1.7). It is beyond the scope of this paper to compare all methods that may be applied to the solution of (1.7). We single out three methods that are easy to implement and that we feel are particularly promising.

The remainder of this paper is organized as follows. Section 2 presents our projection-based nonstationary iterated Tikhonov regularization method for solving the sequence of Tikhonov regularization problems (1.7). The implementation of the algorithm is discussed in Section 3, and Section 4 describes a few numerical examples. A conclusion can be found in Section 5.

## 2 Iterative subspace methods

The normal equations associated with (1.7) are given by

$$(A^T A + \mu_k^{-1} L^T L)x = \mu_k^{-1} L^T L x_{k-1} + A^T b. \tag{2.1}$$

These equations form the basis for our iterative methods. Introduce the subspace $\mathcal{V}_k \subset \mathbb{R}^n$ of small dimension $k \ll n$, and let the columns of the matrix $V_k \in \mathbb{R}^{n\times k}$ form an orthonormal basis for $\mathcal{V}_k$. Substitute $x_k = V_k y_k$ for some $y_k \in \mathbb{R}^k$ into (2.1), and multiply equation (2.1) by $V_k^T$ from the left. We obtain

$$\left((AV_k)^T(AV_k) + \mu_k^{-1}(LV_k)^T(LV_k)\right) y_k = \mu_k^{-1}(LV_k)^T L V_{k-1} y_{k-1} + (AV_k)^T b.$$

The matrix $[AV_k, LV_k]$ is of full rank due to (1.6). Therefore the matrix on the left-hand side is nonsingular for any $0 < \mu_k < \infty$, and we have

$$y_k = \bar{y}_k + \left((AV_k)^T(AV_k) + \mu_k^{-1}(LV_k)^T(LV_k)\right)^{-1} (AV_k)^T \bar{r}_k, \tag{2.2}$$

where

$$\bar{y}_k = \left[y_{k-1}^T, 0\right]^T, \qquad \bar{r}_k = b - (AV_k)\bar{y}_k. \tag{2.3}$$

The iterations are started by choosing a subspace $\mathcal{V}_k$ of dimension $k$ for some small value of $k$, say, $k = 5$. The matrix $V_k$ is in our computed examples determined by the application of $k$ steps of Golub–Kahan bidiagonalization to $A$ with initial vector $b$, or by the application of $k$ steps of the Arnoldi process to $A$ with initial vector $b$. We set $y_{k-1} = 0$ and compute $y_k$ as described above. Then we increase the dimension of the subspace $\mathcal{V}_k$ by one to obtain the subspace $\mathcal{V}_{k+1}$ and determine a new vector $y_{k+1} \in \mathbb{R}^{k+1}$ that satisfies (2.2) with $k$ replaced by $k + 1$. The columns of the matrix $V_{k+1} = [V_k, v_{k+1}] \in \mathbb{R}^{m\times(k+1)}$ form an orthonormal basis for the subspace $\mathcal{V}_{k+1}$, where the vector $v_{k+1}$ is determined by applying one more step of Golub–Kahan bidiagonalization or the Arnoldi process to $A$.

The available bound $\delta$ for the norm of the error $e$ in (1.3) allows us to determine a suitable value of $\mu_k$ in each iteration step with the aid of the discrepancy principle, which prescribes that $\mu_k = \mu_k(\delta)$ be chosen as the zero of the function

$$\varphi(\mu, V_k) := \|Ax_k - b\|^2 - \eta^2\delta^2, \tag{2.4}$$

where $\eta \geq 1$ is a user-specified constant independent of $\delta$. When the bound $\delta$ in (1.3) is fairly sharp and the error $e$ is white Gaussian noise, then the best computed approximations $x_k$ of $x_{\text{true}}$ are generally obtained when $\eta$ is chosen close to unity. We compute the zero of $\varphi(\mu, V_k)$ with a zero-finder described below.

Having determined $\mu_k$, we can compute $y_k$ by (2.2). Then the search subspace $\mathcal{V}_k$ is expanded as described above. After expansion, a new value, $\mu_{k+1}$, of the regularization parameter is calculated by determining the zero of $\varphi(\mu, V_{k+1})$. This process is repeated for increasing $k$-values until two consecutive approximations $x_{k-1} = V_{k-1}y_{k-1}$ and $x_k = V_k y_k$ give residual errors that are sufficiently close. Specifically, let $r_{k-1} = b - AV_{k-1}y_{k-1}$ and $r_k = b - AV_k y_k$. The computations are terminated as soon as the relative residual change satisfies

$$\frac{\|r_k - r_{k-1}\|}{\|b\|} \leq tol \tag{2.5}$$

for some user-specified tolerance $tol$. This stopping criterion is generally satisfied already for a fairly small value of $k$.

Algorithm 2.1 describes a solution method for (1.7) when $A$ is reduced by a few steps of Golub–Kahan bidiagonalization. The algorithm easily can be modified to instead reduce a square matrix $A$ by the Arnoldi process. Most of the computational work is carried out in line 7 of the algorithm, because computing the zero in line 5 and determining $y_k$ by solving a reduced least-squares problem in line 6 are quite inexpensive.

**Algorithm 2.1** A projected nonstationary iterated Tikhonov regularization method

Input: $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $L \in \mathbb{R}^{p\times n}$, $\eta > 1$, and $\delta$.
Determine initial matrix $V_\ell \in \mathbb{R}^{n\times \ell}$ with orthonormal columns by $\ell$ steps of Golub–Kahan bidiagonalization for $\ell > 0$ small. This gives the decomposition $AV_\ell = U_{\ell+1}\tilde{C}_\ell$. Compute the QR factorization $LV_\ell = Q_\ell R_\ell$. Let $\bar{y}_\ell = 0$.
**for** $k = \ell, \ell+1, \ldots$ until convergence **do**
    Compute $\bar{y}_k = \left[y_{k-1}^T, 0\right]^T$ and $\bar{r}_k = b - (AV_k)\bar{y}_k$.
    Find the root $\mu_k$ of $\varphi(\mu, V_k)$ by a zero-finder.
    Compute $y_k$ by solving the reduced least-squares problem (3.5).
    Compute $v_{k+1}$ and $u_{k+1}$ by one step of Golub–Kahan bidiagonalization.
    Let $V_{k+1} = [V_k, v_{k+1}]$ and $U_{k+2} = [U_{k+1}, u_{k+2}]$. This gives the decomposition $AV_{k+1} = U_{k+2}\tilde{C}_{k+1}$.
    Compute the QR factorization $LV_{k+1} = Q_{k+1}R_{k+1}$.
**end for**
Find the root $\mu_{k+1}$ of $\varphi(\mu, V_{k+1})$ by a zero-finder.
Compute $y_{k+1}$ by solving the reduced least-squares problem (3.5) with $k$ replaced by $k+1$.
Determine the approximate Tikhonov solution $x_{k+1} = V_{k+1}y_{k+1}$.

# 3 Some implementation details for Algorithm 2.1

We discuss some details of the implementation of Algorithm 2.1 when the solution subspaces $\mathcal{V}_k$, $k = 1, 2, \ldots$ , are generated by Golub–Kahan bidiagonalization of $A$. Modifications required when the solution subspaces are determined by instead applying a few steps of the Arnoldi process to a square matrix $A$ are straightforward and, therefore, will not be commented on separately.

## 3.1 Golub–Kahan bidiagonalization

Application of $k$ steps of Golub–Kahan bidiagonalization [2, 26] to $A$ with initial vector $b$ gives the decompositions

$$AV_k = U_{k+1}\tilde{C}_k, \qquad A^T U_k = V_k C_k^T, \qquad U_{k+1}e_1 = b/\|b\|, \tag{3.1}$$

where the matrices $U_k = [u_1, u_2, \ldots, u_k] \in \mathbb{R}^{m\times k}$, $U_{k+1} = [U_k, u_{k+1}] \in \mathbb{R}^{m\times(k+1)}$, and $V_k = [v_1, v_2, \ldots, v_k] \in \mathbb{R}^{n\times k}$ have orthonormal columns,

$$C_k = \begin{bmatrix} \alpha_1 & & & & 0 \\ \beta_2 & \alpha_2 & & & \\ & \ddots & \ddots & & \\ & & \beta_{k-1} & \alpha_{k-1} & \\ 0 & & & \beta_k & \alpha_k \end{bmatrix} \in \mathbb{R}^{k\times k} \tag{3.2}$$

is lower bidiagonal, and $\tilde{C}_k = [C_k^T, \beta_{k+1}e_k]^T$. Here $e_j = [0, \ldots, 0, 1, 0, \ldots, 0]^T$ denotes the $j$th axis vector. We assume $k$ to be small enough so that all diagonal and subdiagonal entries of $\tilde{C}_k$ are nonvanishing. The columns of $V_k$ span the Krylov subspace (1.8).

Introduce the QR factorization

$$LV_k = Q_k R_k, \tag{3.3}$$

where $Q_k \in \mathbb{R}^{p\times k}$ has orthonormal columns and $R_k \in \mathbb{R}^{k\times k}$ is upper triangular. The matrix $LV_k$ is not required to be of full rank. Substituting the first part of (3.1), as well as (3.3), into (2.2) produces

$$(\tilde{C}_k^T\tilde{C}_k + \mu_k^{-1}R_k^TR_k)(y_k - \bar{y}_k) = \tilde{C}_k^T(\|b\|e_1 - \tilde{C}_k\bar{y}_k). \tag{3.4}$$

The above are the normal equations associated with the least-squares problem

$$\left\| \begin{bmatrix} \tilde{C}_k \\ \mu_k^{-1/2}R_k \end{bmatrix} (y_k - \bar{y}_k) - \begin{bmatrix} \|b\|e_1 - \tilde{C}_k\bar{y}_k \\ 0 \end{bmatrix} \right\|^2 = \min! \tag{3.5}$$

which we solve for $y_k - \bar{y}_k$. We solve (3.5) instead of (3.4) because of the smaller condition number of the $(2k+1)\times k$ system matrix in (3.5) than of the system matrix in (3.4); see, e.g., Björck [3] for a discussion on the solution of least-squares problems. It follows from (1.6) that the system matrix in (3.5) is nonsingular. The solution $y_k - \bar{y}_k$ yields the desired vector

$$y_k = \bar{y}_k + (\tilde{C}_k^T\tilde{C}_k + \mu_k^{-1}R_k^TR_k)^{-1}\tilde{C}_k^T(\|b\|e_1 - \tilde{C}_k\bar{y}_k).$$

The vector $y_k = y_k(\mu)$ is a function of the regularization parameter $\mu = \mu_k$. We have

$$y_k(\mu) = y(\mu, V_k) = \bar{y}_k + (\tilde{C}_k^T\tilde{C}_k + \mu^{-1}R_k^TR_k)^{-1}\tilde{C}_k^T(\|b\|e_1 - \tilde{C}_k\bar{y}_k).$$

The derivative of $y_k(\mu)$ with respect to $\mu$ is given by

$$y_k'(\mu) = \mu^{-2}(\tilde{C}_k^T\tilde{C}_k + \mu^{-1}R_k^TR_k)^{-1}R_k^TR_k(y_k(\mu) - \bar{y}_k).$$

The quantity $y_k' = y_k'(\mu_k)$ can be evaluated similarly as $y_k$ above, i.e., by solving the least-squares problem

$$\left\| \begin{bmatrix} \tilde{C}_k \\ \mu_k^{-1/2}R_k \end{bmatrix} y_k' - \begin{bmatrix} 0 \\ \mu^{-3/2}R_k(y_k - \bar{y}_k) \end{bmatrix} \right\| = \min! \tag{3.6}$$

Note that the system matrices in (3.5) and (3.6) are the same. Therefore, after having computed the Golub–Kahan bidiagonalization (3.1) and the QR factorization (3.3), we can use these decompositions to solve both least-squares problems (3.5) and (3.6) inexpensively. It is necessary to repeatedly solve these systems for different $\mu_k$-values in order to determine a value such that the associated solution $x_k = V_k y_k$ satisfies the discrepancy principle; see below. The following proposition collects some properties of $y_k(\mu)$ and $y_k'(\mu)$.

**Proposition 3.1** *The following limits hold:*

$$\begin{aligned} y_k(\infty) &:= \lim_{\mu\to\infty} y_k(\mu) = \bar{y}_k + \tilde{C}_k^{\dagger}(\|b\|e_1 - \tilde{C}_k\bar{y}_k), \\ y_k(0) &:= \lim_{\mu\to 0} y_k(\mu) = \bar{y}_k, \\ y_k'(\infty) &:= \lim_{\mu\to\infty} y_k'(\mu) = 0, \\ y_k'(0) &:= \lim_{\mu\to 0} y_\mu' = (R_k^T R_k)^{\dagger}\tilde{C}_k^T(\|b\|e_1 - \tilde{C}_k\bar{y}_k), \end{aligned}$$

*where $M^{\dagger}$ denotes the Moore–Penrose pseudoinverse of the matrix $M$.*

**Proof.** The limits can be shown by fairly straightforward computations. We omit the details since related computations are presented in [18]. □

## 3.2 Properties of the function $\varphi(\mu, V_k)$

We collect some properties of the function (2.4). A related function is described in [18]. The proofs of the properties of the latter presented in [18] are analogous to the corresponding proofs for (2.4). We therefore only outline the proofs for the function (2.4) in this section.

Substituting the factorizations (3.1) and (3.3) into $\varphi(\mu_k, V_k)$, we obtain

$$\begin{aligned} \varphi(\mu_k, V_k) &= \|AV_k y_k - b\|^2 - \eta^2\delta^2 \\ &= \|\tilde{C}_k y_k - U_{k+1}^T b\|^2 - \eta^2\delta^2 \\ &= \|b\|^2 + (\tilde{C}_k y_k)^T(\tilde{C}_k y_k) - 2\|b\|(\tilde{C}_k y_k)^T e_1 - \eta^2\delta^2. \end{aligned}$$

It follows that we can evaluate $\varphi(\mu_k, V_k)$ by solving a small least-squares problem with a $(2k+1)\times k$ matrix consisting of a lower bidiagonal matrix and a triangular matrix; cf. (3.5).

**Theorem 3.2** *Given the GSVD of the matrix pair $\{\tilde{C}_k, R_k\}$,*

$$\tilde{C}_k = \hat{U}_k \Sigma_k Y_k^T, \tag{3.7}$$

$$R_k = \hat{V}_k \Omega_k Y_k^T, \tag{3.8}$$

*where the matrices $\hat{U}_k = [u_1, u_2, \ldots, u_k] \in \mathbb{R}^{(k+1)\times k}$ and $\hat{V}_k \in \mathbb{R}^{k\times k}$ have orthonormal columns, and the nontrivial entries of the matrices*

$$\Sigma_k = \mathrm{diag}[\sigma_1, \sigma_2, \ldots, \sigma_k] \in \mathbb{R}^{k\times k}, \qquad \Omega_k = \mathrm{diag}[\omega_1, \omega_2, \ldots, \omega_k] \in \mathbb{R}^{k\times k}$$

*are ordered according to*

$$0 \le \sigma_1 \le \ldots \le \sigma_k \le 1, \qquad 1 \ge \omega_1 \ge \ldots \ge \omega_k \ge 0$$

*with $\sigma_j^2 + \omega_j^2 = 1$ for $1 \le j \le k$. Moreover, $Y_k \in \mathbb{R}^{k\times k}$ is nonsingular.*

*Let $\bar{r}_k$ be defined by (2.3). If there is an index $i$ such that*

$$((U_{k+1}\hat{U}_k)^T \bar{r}_k)_i \neq 0, \qquad \sigma_i > 0, \qquad \omega_i > 0, \tag{3.9}$$

*then the function $\varphi(\mu, V_k)$ is monotonically decreasing and convex for $\mu > 0$. Moreover, the equation*

$$\varphi(\mu, V_k) = 0 \tag{3.10}$$

*has a unique solution $0 < \mu < \infty$ for any $\eta^2\delta^2$ with*

$$\|\mathrm{P}_{\mathcal{N}(\tilde{C}_k^T)}(U_{k+1}^T \bar{r}_k)\|^2 < \eta^2\delta^2 < \|\bar{r}_k\|^2, \tag{3.11}$$

*where $\mathrm{P}_{\mathcal{N}(\tilde{C}_k^T)}$ denotes the orthogonal projector onto the null space of $\tilde{C}_k^T$.*

**Proof.** We first show monotonicity and convexity. Substituting the factorizations (3.1), (3.3), and (2.2), together with (3.7) and (3.8), into (2.4), we obtain after some calculations

$$\begin{aligned} & \varphi(\mu, V_k) + \eta^2\delta^2 = \|AV_k(y_k - \bar{y}_k) - (b - AV_k\bar{y}_k)\|^2 \\ = \;& \|U_{k+1}\tilde{C}_k(\tilde{C}_k^T\tilde{C}_k + \mu^{-1}R_k^T R_k)^{-1}(U_{k+1}\tilde{C}_k)^T\bar{r}_k - \bar{r}_k\|^2 \\ = \;& \sum_{i=1}^{k} \frac{\omega_i^4}{(\mu\sigma_i^2 + \omega_i^2)^2}((U_{k+1}\hat{U}_k)^T\bar{r}_k)_i^2 + (U_{k+1}^T\bar{r}_k)^T(I - \hat{U}_k\hat{U}_k^T)U_{k+1}^T\bar{r}_k. \end{aligned}$$

It follows that the derivative of $\varphi$ with respect to $\mu$ is given by

$$\varphi'(\mu, V_k) = -\sum_{i=1}^{k} \frac{2\sigma_i^2\omega_i^4}{(\mu\sigma_i^2 + \omega_i^2)^3}((U_{k+1}\hat{U}_k)^T\bar{r}_k)_i^2,$$

and the second derivative is

$$\varphi''(\mu, V_k) = \sum_{i=1}^{k} \frac{6\sigma_i^4\omega_i^4}{(\mu\sigma_i^2 + \omega_i^2)^4}((U_{k+1}\hat{U}_k)^T\bar{r}_k)_i^2.$$

We note that $\sigma_i + \omega_i > 0$ for all $i$. The condition (3.9) secures that $\varphi'(\mu, V_k) < 0$ and $\varphi''(\mu, V_k) > 0$. Thus, $\varphi(\mu, V_k)$ is monotonically decreasing and convex for $\mu > 0$ on the subspace $\mathcal{V}_k$.

Turning to the solubility of equation (3.10), we observe that

$$\begin{aligned} & \lim_{\mu\to 0} \varphi(\mu, V_k) + \eta^2\delta^2 = \lim_{\mu\to 0} \|AV_k y_k - b\|^2 \\ = \;& \lim_{\mu\to 0} \|(\tilde{C}_k(\tilde{C}_k^T\tilde{C}_k + \mu^{-1}R_k^T R_k)^{-1}R_k^T - I)U_{k+1}^T\bar{r}_k\|^2 \\ = \;& \|U_{k+1}^T\bar{r}_k\|^2 = \|\bar{r}_k\|^2 \end{aligned} \tag{3.12}$$

and

$$\begin{aligned} & \lim_{\mu\to\infty} \varphi(\mu, V_k) + \eta^2\delta^2 \\ = \;& \lim_{\mu\to\infty} \|(\tilde{C}_k(\tilde{C}_k^T\tilde{C}_k + \mu^{-1}R_k^T R_k)^{-1}\tilde{C}_k^T - I)U_{k+1}^T\bar{r}_k\|^2 \\ = \;& (U_{k+1}^T\bar{r}_k)^T(I - \tilde{C}_k\tilde{C}_k^\dagger)U_{k+1}^T\bar{r}_k. \end{aligned} \tag{3.13}$$

Since $\varphi(\mu, V_k)$ is monotonically decreasing and convex for any $\mu > 0$ on the subspace $\mathcal{V}_k$, the above limits imply that equation (3.10) has a unique solution when (3.11) holds. This completes the proof. $\square$

We remark that the convexity of $\varphi(\mu, V_k)$ is a consequence of using $1/\mu$ in (1.5) instead of $\mu$. The proposition below provides some asymptotic properties of the function $\varphi(\mu, V_k)$ and its derivatives.

**Proposition 3.3** *The following limits hold:*

$$\begin{aligned}
\lim_{\mu\to 0} \varphi(\mu, V_k) &= \|\bar{r}_k\|^2 - \eta^2\delta^2, && (3.14)\\
\lim_{\mu\to \infty} \varphi(\mu, V_k) &= \|\bar{r}_k\|^2 - \|P_{\mathcal{R}(\tilde{C}_k^T)}(\|b\|e_1 - \tilde{C}_k\bar{y}_k)\|^2 - \eta^2\delta^2, && (3.15)\\
\lim_{\mu\to 0} \varphi'(\mu, V_k) &= -2\|(R_k)^\dagger \tilde{C}_k^T(\|b\|e_1 - \tilde{C}_k\bar{y}_k)\|^2.\\
\lim_{\mu\to \infty} \varphi'(\mu, V_k) &= 0,\\
\lim_{\mu\to 0} \varphi''(\mu, V_k) &= 6\|\tilde{C}_k(R_k^T R_k)^\dagger \tilde{C}_k^T(\|b\|e_1 - \tilde{C}_k\bar{y}_k)\|^2,\\
\lim_{\mu\to \infty} \varphi''(\mu, V_k) &= 0.
\end{aligned}$$

**Proof.** The limits (3.14) and (3.15) follow from (3.12) and (3.13). The remaining limits can be shown by using the corresponding limits for $y_k(\mu)$. $\square$

Since $\varphi(\mu, V_k)$ is monotonically decreasing and convex for $\mu > 0$ under the condition that there is an index $i$ so that (3.9) holds, the limit (3.15) has to be negative in order for $\varphi$ to have a finite zero. Therefore, if $\|\bar{r}_k\|^2 - \|P_{\mathcal{R}(\tilde{C}_k^T)}(\|b\|e_1 - \tilde{C}_k\bar{y}_k)\|^2 > \eta^2\delta^2$ for the initial matrix $V_k$, the initial solution subspace has to be enlarged to secure that $\varphi$ has a finite zero. In general, $\|P_{\mathcal{N}(\tilde{C}_k^T)}(\|b\|e_1 - \tilde{C}_k\bar{y}_k)\|^2 \ll \eta^2\delta^2$ and $\varphi(\mu, V_k)$ has a finite zero already for solution subspaces of small dimension $k$. We remark that if $\lim_{\mu\to\infty} \varphi(\mu, V_k) < 0$ for some $k \geq 1$, then this inequality also holds for all larger values of $k$.

## 3.3 Determining the regularization parameter

We use a zero-finder based on inverse rational interpolation to determine the zero of the function $\varphi(\mu, V_k)$ defined by (2.4). This zero-finder is advocated by Lampe et al. [20]. We provide a brief outline and describe formulas that differ from those used in [20]. In this section, we use the simplified notation $f(\mu) = \varphi(\mu, V_k)$ and similarly for the derivatives.

We remark that a simple way to evaluate the functions $f(\mu)$, $f'(\mu)$, and $f''(\mu)$ is to first compute the GSVD of the matrix pair $\{AV_k, LV_k\}$. However, there is no inexpensive way to update the GSVD of $\{AV_k, LV_k\}$ to obtain the GSVD of $\{AV_{k+1}, LV_{k+1}\}$. We therefore seek to determine the zeros of $\varphi(\mu, V_k)$ for increasing values of $k$ without using the GSVD.

The zero-finder advocated in [20] uses a rational function $h(f)$ to approximate $f^{-1}$. The function $h(f)$ is determined by using values of $f$ and $f'$ at

two points, which we denote by $\mu^1$ and $\mu^2$. The next iterate is $\mu_{\text{new}} = h(0)$. Specifically, we use the model function

$$h(f) = \frac{p(f)}{f - f_\infty}$$

for approximating $f^{-1}$. Here $p(t) = \sum_{i=0}^{3} a_i t^i$ and

$$f_\infty = \|\bar{r}_k\|^2 - (\|b\|e_1 - \tilde{C}_k \bar{y}_k)^T (\tilde{C}_k \tilde{C}_k^\dagger)(\|b\|e_1 - \tilde{C}_k \bar{y}_k) - \eta^2 \delta^2.$$

The coefficients $a_i$ are determined by the interpolation conditions

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

with the points $\mu^1$ and $\mu^2$ chosen so that the function values $f(\mu^1)$ and $f(\mu^2)$ do not have the same sign. We remark that 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 computation of the coefficients $a_i$ of the polynomial $p$ requires the solution of a linear system of equations with a $4 \times 4$ matrix. To improve the conditioning of this matrix, we use a basis of Chebyshev polynomials for the interval between the points $f(\mu^1)$ and $f(\mu^2)$ instead of the power basis in the computations.

The value $\mu_{\text{new}}$ replaces the value $\mu^i$ 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 $\mu^i$ is updated.

When the requirement (3.11) on the solution subspace $\mathcal{V}_k$ is violated, it is not possible to find a zero of $f(\mu)$ (except when $f(\mu) \equiv 0$). This situation may arise, if we choose the initial search space $\mathcal{V}_k$ to be a subset of $\mathcal{N}(L)$. Then the matrix $LV_k$ is the zero-matrix. A simple way to handle this situation is to increase the dimension of the initial solution subspace $\mathcal{V}_k$, e.g., by carrying out a few more Golub–Kahan bidiagonalization steps, until (3.11) holds.

## 3.4 Enlarging the Search Space

We discuss how the solution subspaces are enlarged and describe some updating formulas that are convenient to use. In the beginning of this subsection, we will assume that the solution subspaces are determined by partial Golub–Kahan bidiagonalization of $A$. Subsequently, we will discuss the situation when the solution subspaces are generated by the Arnoldi process.

The recursion formulas for Golub–Kahan bidiagonalization of $A$ with initial vector $b$ are given by

$$\begin{array}{rclcrcl} \beta_1 u_1 & = & b, & & \alpha_1 v_1 & = & A^T u_1; \\ \beta_{k+1} u_{k+1} & = & A v_k - \alpha_k u_k, & & \alpha_{k+1} v_{k+1} & = & A^T u_{k+1} - \beta_{k+1} u_k, \end{array} \tag{3.16}$$

for $k = 1, 2, \ldots$ . The scalars $\alpha_k, \beta_k$, $k = 1, 2, \ldots$ , are chosen so that the vectors $u_k$ and $v_k$ are of unit length. These vectors form the orthonormal columns of the matrices $U_k$ and $V_k$ in (3.1), and the scalars $\alpha_k, \beta_k$ are entries of the matrices (3.2) and $\tilde{C}_k$; see [2, 26] for details. We will use the recursion formulas (3.16) to determine new columns to be appended to the matrices $V_k$ and $U_k$. Thus, we define $V_{k+1} = [V_k, v_{k+1}]$ and $U_{k+1} = [U_k, u_{k+1}]$. Similarly, we update $C_k$ and $\tilde{C}_k$ to obtain $C_{k+1}$ and $\tilde{C}_{k+1}$. The main computational cost is the evaluation of two matrix-vector products; one with $A$ and one with $A^T$.

When enlarging the solution subspace in Algorithm 2.1 by the new vector $v_{k+1}$, we update the matrices $AV_k$ and $LV_k$ to obtain $AV_{k+1}$ and $LV_{k+1}$ by appending $Av_{k+1}$ and $Lv_{k+1}$ as the last columns of the matrices $AV_k$ and $LV_k$, respectively. These computations cost only one matrix-vector product evaluation with each one of the matrices $A$ and $L$. Matrices $L$ used in applications typically approximate derivative operators and therefore are very sparse. The computational work required for evaluating matrix-vector products with $L$ therefore generally can be ignored. Moreover, the matrix-vector product $Av_{k+1}$ is used in the next step of Golub–Kahan bidiagonalization. Therefore a typical iteration step only requires one matrix-vector product evaluation with each one of the matrices $A$ and $A^T$.

The QR factorization of $LV_k = Q_k R_k$ is updated as follows:

$$LV_{k+1} = L[V_k, v_{k+1}] \quad = \quad [Q_k, \tilde{q}_{k+1}] \begin{bmatrix} R_k & r_{k+1} \\ 0 & \rho_{k+1} \end{bmatrix}, \tag{3.17}$$

where $Q_{k+1} = [Q_k, \tilde{q}_{k+1}] \in \mathbb{R}^{p\times(k+1)}$ has orthonormal columns, $r_{k+1} \in \mathbb{R}^k$, and $\rho_{k+1} \in \mathbb{R}$; see [6] for a detailed discussion on updating and downdating of the QR factorization. The vector $r_{k+1}$ and scalar $\rho_{k+1}$ are determined by

$$\begin{array}{rclcrcl} r_{k+1} & = & Q_k^T(Lv_{k+1}), & & q_{k+1} & = & Lv_{k+1} - r_{k+1}, \\ \rho_{k+1} & = & \|q_{k+1}\|, & & \tilde{q}_{k+1} & = & q_{k+1}/\rho_{k+1}. \end{array}$$

In case $\rho_{k+1} = 0$, the vector $\tilde{q}_{k+1}$ is chosen as an arbitrary unit vector such that $Q_k^T \tilde{q}_{k+1} = 0$. The main computational cost for determining the decomposition (3.17) is the evaluation of the matrix-vector product with $Q_k^T \in \mathbb{R}^{k\times p}$. Since $k$ is quite small, the cost for updating the factorization (3.17) is negligible.

We turn to the situation when the solution subspace is generated by the Arnoldi process and assume that $A \in \mathbb{R}^{n\times n}$. Application of $k$ steps of the Arnoldi process to $A$ with initial vector $b$ gives the decomposition

$$AV_k = V_{k+1}\tilde{H}_k, \tag{3.18}$$

where $V_k e_1 = b/\|b\|$ and the orthonormal columns of the matrix $V_k \in \mathbb{R}^{n\times k}$ forms a basis for the Krylov subspace (1.9) with $q = 0$. The columns of $V_{k+1} = [V_k, v_{k+1}] \in \mathbb{R}^{n\times(k+1)}$ are also orthonormal. The matrix $\tilde{H}_k \in \mathbb{R}^{(k+1)\times k}$ is upper Hessenberg with positive subdiagonal entries. We assume here that $k$ is small enough so that breakdown of the recursions for the Arnoldi process does not occur. This is the generic situation. Breakdown is a rare event that

typically simplifies the computations; see Saad [29] for details on the Arnoldi process. The decomposition (3.18) can be used similarly as the Golub–Kahan bidiagonalization in Algorithm 2.1. To increase the number of columns of $V_k$ by one, one more step of the Arnoldi process is executed. This gives formulas analogous to (3.16). The QR factorization is updated as described by (3.17).

In the generalized Krylov subspace (GKS) method, the solution subspace is enlarged in each step by including the residual vector for the normal equations,

$$r_k = A^T A V_k y_k + \mu_k^{-1} L^T L V_k y_k - \mu_k^{-1} L^T L x_{k-1} - A^T b,$$

in the present solution subspace, cf. (2.1). We use the MATLAB function reorth from [16] to insure that $r_k$ is orthogonal to the columns of $V_k$ and then define $V_{k+1} = [V_k, r_k/\|r_k\|]$. This expansion approach was first used by Voss [31] in a method for solving nonlinear eigenvalue problems. More recent applications are described in [20, 21, 22].

# 4 Numerical Examples

Several solution methods for projected nonstationary iterated Tikhonov regularization are compared in order to gain some insight into the importance of the choice of solution subspace. We refer to results achieved with Algorithm 2.1 as PNITR-GKB. The solution subspace is for this method determined by partial Golub–Kahan bidiagonalization. When this bidiagonalization method is replaced by the Arnoldi process, as described in Subsection 3.4, we obtain the PNITR-ARN method. Recently, a projected nonstationary iterated Tikhonov regularization method, in which the solution subspace is enlarged by including the current residual vector for the normal equations (2.1), was proposed in [18]. This expansion approach leads to solution subspaces that are generalized Krylov subspaces (GKS). We refer to this method as PNITR-GKS. This expansion approach previously has been applied in a projected (noniterated) Tikhonov regularization method [20], which we below refer to as PTR-GKS.

We illustrate the performance of these methods when applied to three test examples, the first two of which are from Hansen's Regularization Tools [16]. The last example is concerned with the restoration of a blurred and noisy image. The matrices in all examples are numerically singular. We determine the regularization parameter in all examples by the discrepancy principle. In the first two examples we set $\eta = 1.1$ in (2.4) and $tol = 1 \cdot 10^{-6}$ in the stopping criterion (2.5). All computations were carried out in MATLAB version 2012a with about 15 significant decimal digits on a computer with an Intel Core i5-3230M @ 2.60GHz processor and 8GB RAM.

In the examples $x_{\text{true}}$ is provided, from which we compute $b_{\text{true}} := Ax_{\text{true}}$. White Gaussian noise $e$ is added to $b_{\text{true}}$. This gives 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.

Table 1: Example 4.1: Comparison of relative errors using PTR-GKS, PNITR-GKS, PNITR-GKB, and PNITR-ARN for different regularization matrices.

| Method | $L = L_1$ | $L = L_2$ |
|---|---|---|
| PTR-GKS | 0.056 | 0.061 |
| PNITR-GKS | 0.058 | 0.058 |
| PNITR-GKB | 0.048 | 0.049 |
| PNITR-ARN | 0.048 | 0.049 |

**Example 4.1.** Consider the Fredholm integral equation of the first kind

$$\int_{-\pi/2}^{\pi/2} \kappa(t,s)x(s)ds = b(t), \qquad -\pi/2 \leq t \leq \pi/2, \tag{4.1}$$

with the kernel

$$\kappa(t,s) = (\cos(s) + \sin(t))^2 (\frac{\sin(\xi)}{\xi})^2, \qquad \xi = \pi(\sin(s) + \cos(t)).$$

The right-hand side function $b(t)$ is chosen so that the solution $x(s)$ is the sum of two Gaussian functions. This integral equation is discussed by Shaw [30]. We use the MATLAB code shaw from [16] to discretize (4.1) by a Galerkin method with 1000 orthonormal box functions as test and trial functions. The code produces a matrix $A \in \mathbb{R}^{1000\times 1000}$ and the solutions $x_{\text{true}}$. We compute $b_{\text{true}}$ and the error-contaminated vector $b$ with noise level $\sigma = 1 \cdot 10^{-3}$ as described above.

The condition number $\kappa(A) := \|A\|\|A^\dagger\|$ of $A$, as computed with the MATLAB function cond, is larger than $1.1 \cdot 10^{21}$. Thus, the matrix is numerically singular. The regularization matrix $L$ is chosen to be a scaled discrete first or second order derivative operator in one space-dimension, i.e., the bidiagonal matrix

$$L_1 = \begin{bmatrix} 1 & -1 & & \\ & \ddots & \ddots & \\ & & 1 & -1 \end{bmatrix} \in \mathbb{R}^{(n-1)\times n}, \tag{4.2}$$

or the tridiagonal matrix

$$L_2 = \begin{bmatrix} -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \end{bmatrix} \in \mathbb{R}^{(n-2)\times n}, \tag{4.3}$$

with $n = 1000$.

The initial solution subspace is of dimension six. The PNITR-GKS and PTR-GKS methods use the same initial solution subspace as PNITR-GKB. We

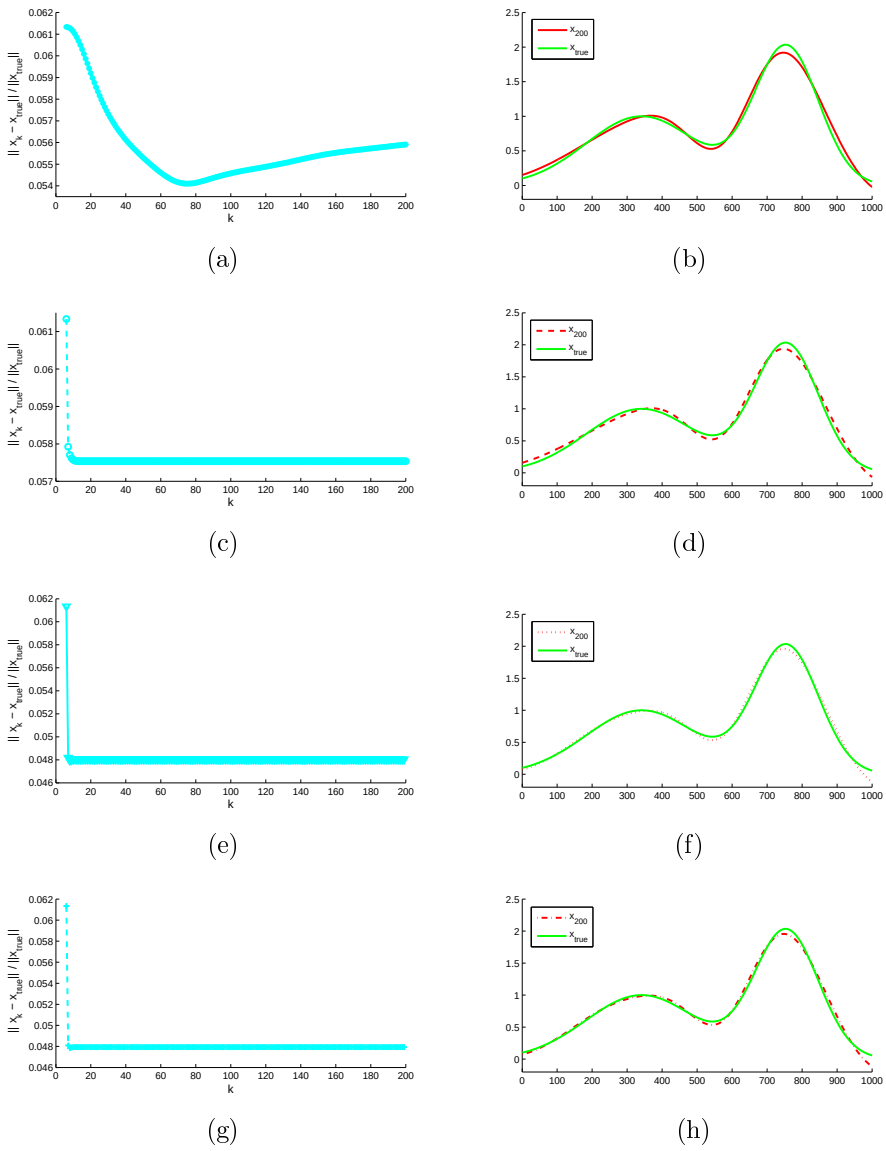



Figure 1: Example 4.1: Relative errors (left column), and exact and computed approximate solutions (right column) by PTR-GKS (first row), PNITR-GKS (second row), PNITR-GKB (third row), and PNITR-ARN (fourth row) for $L = L_1$.

would like to illustrate the convergence of several interesting quantities and therefore do not terminate the computations until the solution subspace has dimension 200. Table 1 compares relative errors using several methods and

the regularization matrices $L_1$ in (4.2) and $L_2$ in (4.3). The table shows all projected nonstationary iterated Tikhonov regularization methods to give more accurate approximations of $x_{\text{true}}$ than the noniterated Tikhonov regularization method PTR-GKS. Moreover, both Krylov subspace methods PNITR-GKB and PNITR-ARN yield approximations of higher quality than PNITR-GKS, which uses a generalized Krylov subspace as solution subspace.

The left-hand side column of Figure 1 displays the relative errors $\{\|x_k - x_{\text{true}}\|/\|x_{\text{true}}\|\}$ for approximate solutions $x_k$, $1 \leq k \leq 200$, determined by the regularization methods PTR-GKS, PNITR-GKS, PNITR-GKB, and PNITR-ARN for $L = L_1$. The right-hand side column shows the exact solution $x_{\text{exact}}$ (green graph) and the computed approximations $x_{200}$ by the different methods according to Table 1. The errors in the computed solutions for the projected nonstationary iterated Tikhonov regularization methods quickly decay with increasing iteration number $k$ and do not vary significantly when $k$ is increased further. This implies that the exact choice of the number of iterations $k$ is not important for the accuracy in the computed approximations $x_k$ as long as $k$ is sufficiently large. The relative errors in the approximate solutions $x_k$ computed with PTR-GKS method reach a minimum faster than for the projected nonstationary iterated methods, but then increase before plateauing. It is difficult to determine the number of iterations $k$ that give the smallest error in $x_k$. The behavior of the solution methods is similar when $L = L_2$. We therefore omit the graphs for this regularization matrix. □

Table 2: Example 4.2: Comparison of relative errors using PTR-GKS, PNITR-GKS, PNITR-GKB, and PNITR-ARN for different regularization matrices.

| Method | $L = L_1$ | $L = L_2$ |
|---|---|---|
| PTR-GKS | 0.014 | 0.015 |
| PNITR-GKS | 0.013 | 0.014 |
| PNITR-GKB | 0.011 | 0.012 |
| PNITR-ARN | 0.013 | 0.013 |

**Example 4.2.** We investigate the performance of the regularization methods when applied to the inverse heat equation problem implemented by the MATLAB code `heat` from [16] with parameter $\kappa = 5$. This code determines the matrix $A \in \mathbb{R}^{1000\times 1000}$ and exact solution $x_{\text{true}} \in \mathbb{R}^{1000}$. We compute $b_{\text{true}} := Ax_{\text{true}}$ and add Gaussian noise of noise level $1 \cdot 10^{-2}$ to obtain the error-contaminated vector $b$. The condition number $\kappa(A) := \|A\|\|A^\dagger\|$ is larger than $3.5 \cdot 10^{19}$. Thus, the matrix is numerically singular. We would like to illustrate the convergence of several interesting quantities and therefore do not terminate the computations until the solution subspace has dimension 100, similarly as in Example 4.1.

Table 2 reports the relative errors of the computed solution obtained with the regularization matrices (4.2) and (4.3). The table shows the regularization matrix $L_1$ to give approximate solutions of slightly higher quality than $L_2$, and

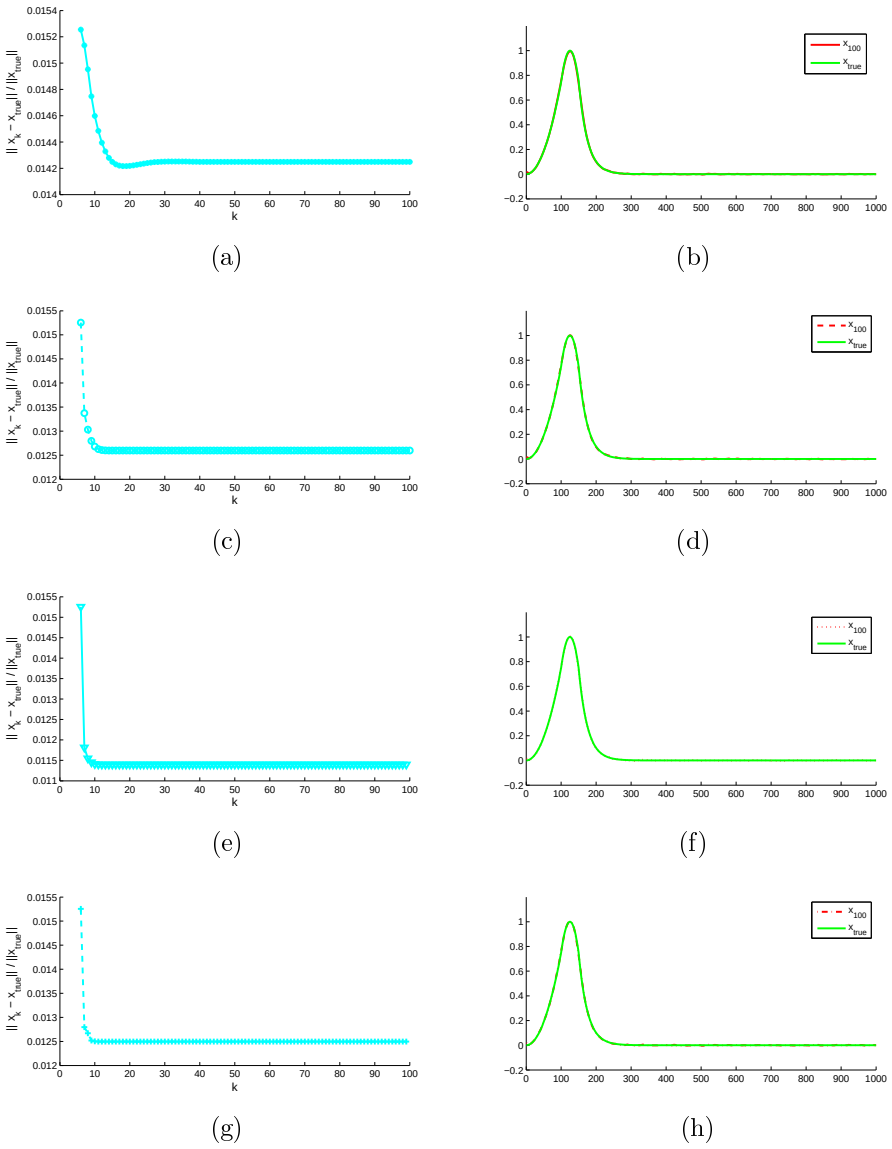



Figure 2: Example 4.2: Relative errors (left column), exact and computed approximate solutions (right column) by PTR-GKS (first row), PNITR-GKS (second row), PNITR-GKB (third row), and PNITR-ARN (fourth row) for $L = L_1$.

the projected nonstationary iterated Tikhonov methods to give iterates of higher accuracy than the PTR-GKS method.

Figure 2 is analogous to Figure 1 and shows results for the regularization

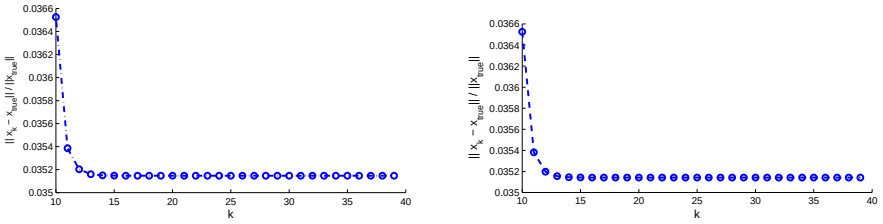
matrix $L_1$. The figure shows the relative errors in the iterates $x_k$ to become small faster for the PNITR methods than for PTR-GKS, and the errors of iterates generated by the latter method plateau at a higher level than the errors of iterates computed by PNITR methods. □

(a) (b)

Figure 3: Example 4.3: (a) Original image "Opfer", (b) blur and noise contaminated image.

Table 3: Example 4.3: Comparisons of relative residual norm and relative errors corresponding when restoring Figure 3(b) using PNITR methods with $L = L_{1,2D}$.

| Relative residual norm | |
|---|---|
| PNITR-GKS | 0.0095 |
| PNITR-GKB | 0.0096 |
| PNITR-ARN | 0.0096 |
| Relative error | |
| PNITR-GKS | 0.035 |
| PNITR-GKB | 0.035 |
| PNITR-ARN | 0.035 |




(a) (b)

Figure 4: Example 4.3: Comparison of relative errors of the restored images using (a) PNITR-GKB and (b) PNITR-ARN with different $L = L_{1,2D}$.

**Example 4.3.** This example discusses the restoration of a gray-scale image that is represented by an array of $256 \times 256$ pixels. Figure 3(a) shows the original image "Opfer", which is assumed to be unknown. The pixel values are ordered column-wise and stored in $x_{\text{true}} \in \mathbb{R}^{65536}$. Figure 3(b) displays a blur- and noise-contaminated version of the original image that we would like to restore. The pixel values for this image are stored in $b \in \mathbb{R}^{65536}$. The blur is Gaussian; the blurring matrix $A \in \mathbb{R}^{65536 \times 65536}$ is block Toeplitz with Toeplitz blocks. It is generated with the MATLAB function blur from [16] using the parameter values band $= 5$ and sigma $= 1.0$. Let $b_{\text{true}} \in \mathbb{R}^{65536}$. The noise is white Gaussian with noise level $1 \cdot 10^{-2}$ and is stored in $e \in \mathbb{R}^{65536}$. The vector $b$ is determined by (1.2). The factor $\eta$ in (2.4) is set to 1.05 and we choose the regularization matrix

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

where $L_1$ is defined by (4.2) with $n = 256$, and $\otimes$ denotes Kronecker product.

We would like to determine an approximation of $x_{\text{true}}$ given $A$ and $b$. To achieve this, we carry out 40 steps with each one of the PNITR-GKB, PNITR-ARN, and PNITR-GKS methods. The limit on the number of steps is dictated by the large size of the solution subspace basis vectors that have to be stored. Table 3 displays the relative residual norms and relative solution errors of the images determined with these method, and Figure 4 illustrates the convergence of the relative errors. The difference in quality of the restored images is insignificant. We note that since PNITR-ARN does not require matrix-vector product evaluations with $A^T$, the method only computes half the number of matrix-vector products as PNITR-GKB and PNITR-GKS. The restored image determined by PNITR-ARN is shown in Figure 5.

# 5 Conclusion

This paper proposes to choose standard Krylov subspaces for projected nonstationary iterated Tikhonov regularization of large-scale problems. The method

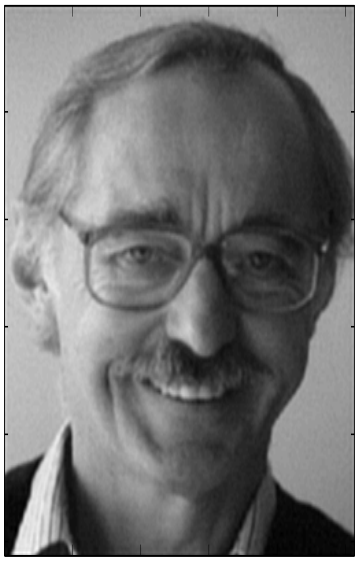


Figure 5: Example 4.3: Restored image "Opfer" with the PNITR-ARN method.

based on the Arnoldi process is particularly attractive because it requires fewer matrix-vector product evaluations than the other PNITR methods in our comparison and yields computed solution of competitive quality.

## Acknowledgments

Work of G.H. was supported by Fund of China Scholarship Council, the young scientific research backbone teachers of CDUT (KYGG201309), work of L.R. was supported in part by NSF grant DMS-1115385, and work of F.Y. was partly supported by grant NNSFC 11501392, Key Natural Science Foundation of Sichuan Education Department grant 15ZA0234, and key projects of SUSE grants 2014QZY01 and 2015RC08. The authors would like to thank a referee for comments.

## References

[1] D. Bianchi, A. Buccini, M. Donatelli, and S. Serra–Capizzano, Iterated fractional Tikhonov regularization, Inverse Problems, 31 (2015), 055005.

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

[3] Å. Björck, Numerical Methods in Matrix Computations, Springer, New York, 2015.

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

[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 (2000), pp. 423–446.

[6] 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 (1976), pp. 772–795.

[7] M. Donatelli and M. Hanke, Fast nonstationary preconditioned iterative methods for ill-posed problems, with application to image deblurring, Inverse Problems, 29 (2013), 095008.

[8] L. Dykes, S. Noschese, and L. Reichel, Rescaling the GSVD with application to ill-posed problems, Numer. Algorithms, 68 (2015), pp. 531–545.

[9] L. Dykes and L. Reichel, A family of range restricted iterative methods for linear discrete ill-posed problems, Dolomites Research Notes on Approximation, 6 (2013), pp. 27–36.

[10] L. Eldén, A weighted pseudoinverse, generalized singular values, and constrained least squares problems, BIT, 22 (1982), pp. 487–502.

[11] H. W. Engl, On the choice of the regularization parameter for iterated Tikhonov regularization of ill-posed problems, J. Approx. Theory, 49 (1987), pp. 55–63.

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

[13] S. Gazzola, P. Novati, and M. R. Russo, On Krylov projection methods and Tikhonov regularization, Electron. Trans. Numer. Anal., 44 (2015), pp. 83–123.

[14] M. Hanke and C. W. Groetsch, Nonstationary iterated Tikhonov regularization, J. Optim. Theory Appl., 436 (1998), pp. 37–53.

[15] P. C. Hansen, Rank-Deficient and Discrete Ill-Posed Problems, SIAM, Philadelphia, 1998.

[16] P. C. Hansen, Regularization tools version 4.0 for Matlab 7.3, Numer. Algorithms, 46 (2007), pp. 189–194.

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

[18] G. Huang, L. Reichel, and F. Yin, Projected nonstationary iterated Tikhonov regularization, BIT, in press.

[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 (2007), pp. 315–330.

[20] J. Lampe, L. Reichel, and H. Voss, Large-scale Tikhonov regularization via reduction by orthogonal projection, Linear Algebra Appl., 436 (2012), pp. 2845–2865.

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

[22] J. Lampe and H. Voss, Large-scale dual regularized total least squares, Electron. Trans. Numer. Anal., 42 (2014), pp. 13–40.

[23] S. Morigi, L. Reichel, and F. Sgallari, Orthogonal projection regularization operators, Numer. Algorithms, 44 (2007), pp. 99–114.

[24] A. Neuman, L. Reichel, and H. Sadok, Implementations of range restricted iterative methods for linear discrete ill-posed problems, Linear Algebra Appl., 436 (2012), pp. 3974–3990.

[25] 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 (1981), pp. 474–489.

[26] C. C. Paige and M. A. Saunders, LSQR: an algorithm for sparse linear equations and sparse least squares problems, ACM Trans. Math. Software, 8 (1982), pp. 43–71.

[27] L. Reichel, F. Sgallari, and Q. Ye, Tikhonov regularization based on generalized Krylov subspace methods, Appl. Numer. Math., 62 (2012), pp. 474–489.

[28] L. Reichel and X. Yu, Matrix decompositions for Tikhonov regularization, Electron. Trans. Numer. Anal., 43 (2015), pp. 223–243.

[29] Y. Saad, Iterative Methods for Sparse Linear Systems, SIAM, Philadelphia, 2003.

[30] C. B. Shaw, Jr., Improvements of the resolution of an instrument by numerical solution of an integral equation, J. Math. Anal. Appl., 37 (1972), pp. 83–112.

[31] H. Voss, An Arnoldi method for nonlinear eigenvalue problems, BIT, 44 (2004), pp. 387–401.