SIAM J. Sci. Comput.
Vol. 46, No. 4, pp. A2607–A2633

A PRECONDITIONED KRYLOV SUBSPACE METHOD FOR LINEAR INVERSE PROBLEMS WITH GENERAL-FORM TIKHONOV REGULARIZATION[*]

HAIBO LI[†]

Abstract. Tikhonov regularization is a widely used technique in solving inverse problems that can enforce prior properties on the desired solution. In this paper, we propose a Krylov subspace based iterative method for solving linear inverse problems with general-form Tikhonov regularization term $x^T Mx$, where M is a positive semidefinite matrix. An iterative process called the preconditioned Golub–Kahan bidiagonalization (pGKB) is designed, which implicitly utilizes a proper preconditioner to generate a series of solution subspaces with desirable properties encoded by the regularizer $x^T Mx$. Based on the pGKB process, we propose an iterative regularization algorithm via projecting the original problem onto small dimensional solution subspaces. We analyze the regularization properties of this algorithm, including the incorporation of prior properties of the desired solution into the solution subspace and the semiconvergence behavior of the regularized solution. To overcome instabilities caused by semiconvergence, we further propose two pGKB based hybrid regularization algorithms. All the proposed algorithms are tested on both small-scale and large-scale linear inverse problems. Numerical results demonstrate that these iterative algorithms exhibit excellent performance, outperforming other state-of-the-art algorithms in some cases.

Key words. inverse problems, ill-posed, general-form Tikhonov regularization, preconditioned Golub-Kahan bidiagonalization, subspace projection regularization, hybrid regularization

MSC codes. 65F22, 65J20, 65J22

DOI. 10.1137/23M1593802

1. Introduction.

Inverse problems arise in various fields of science and engineering, where the aim is to recover unknown parameters or functions from observed data. Such problems are often encountered in many applications, including image reconstruction, medical imaging, geophysics, data assimilation, and so on [6, 28, 35, 38, 48]. Formally, a linear inverse problem after discretization leads to the following linear system,

$$Ax_{\text{true}} + e = b, \tag{1.1}$$

where $A \in \mathbb{R}^{m\times n}$ is the (discretized) forward operator that maps the unknown quantity to the observed data, e is the noise in the observed data, and x_{true} is the underlying quantity we wish to reconstruct. One key issue with inverse problems is that they are usually ill-posed. For (1.1) it means that A is extremely ill-conditioned such that a small perturbation in b leads to large changes in the solution, or A is underdetermined such that there may be multiple solutions that fit the data equally well [15, 25]. These difficulties stem from the fact that the inverse of the forward operator is usually discontinuous or fails to preserve certain properties of the desired solution, such as smoothness or sparsity [15].

To overcome these challenges, regularization techniques are commonly employed, which use prior knowledge about the underlying solution to constrain the set of possible solutions and improve their stability and uniqueness properties. The idea is to

[*]Submitted to the journal's Numerical Algorithms for Scientific Computing section August 14, 2023; accepted for publication (in revised form) May 22, 2024; published electronically August 14, 2024.
https://doi.org/10.1137/23M1593802

[†]School of Mathematics and Statistics, The University of Melbourne, Parkville, VIC, 3010, Australia (haibolee1729@gmail.com, haibo.li@unimelb.edu.au).

introduce a penalty term into the objective function to promote solutions that are smooth, sparse, or have other desirable properties [15, 27]. Tikhonov regularization seeks to solve the following regularized inverse problem:

$$x_\lambda = \underset{x\in\mathbb{R}^n}{\arg\min}\{\|Ax-b\|_2^2+\lambda R(x)\}, \tag{1.2}$$

where $\lambda > 0$ is a regularization parameter that controls the trade-off between data fit and regularization, and $R(x)$ is a regularizer that encodes our prior knowledge about the solution. A popular choice of the regularizer is $R(x)=\|Lx\|_2^2$ with a linear operator $L\in\mathbb{R}^{p\times n}$ that maps x to a suitable space [52]. Regularization based on Bayesian inference is another commonly used regularization method [35, 51]. Suppose $e\sim\mathcal{N}(0,\mu^{-1}I)$ is a Gaussian random vector. In the paper we denote by I the identity matrix with order clear from the context. Then the likelihood of $b|x$ satisfies $\pi(b|x)\propto \exp\left(-\frac{\mu}{2}(Ax-b)^T(Ax-b)\right)$. If we choose a Gaussian prior $\pi(x)\propto\exp\left(-\frac{\sigma}{2}x^TMx\right)$ to model the distribution of x, where M is positive semidefinite, by the Bayes' formula we have the posterior likelihood:

$$\pi(x|b)\propto\pi(x)\pi(b|x)\propto\exp\left(-\frac{\mu}{2}(Ax-b)^T(Ax-b)-\frac{\sigma}{2}x^TMx\right).$$

By neglecting the scaling factor $\mu/2$, the maximum a posterior estimate of x is the solution to

$$\min_{x\in\mathbb{R}^n}\left\{\|Ax-b\|_2^2+\frac{\sigma}{\mu}x^TMx\right\}. \tag{1.3}$$

Comparing (1.3) with (1.2), we know that σ/μ plays the role of the regularization parameter λ, and if $M=L^TL$ is the square root decomposition of M (note in some literature the square root of M requires $L^T=L$), then $x^TMx=\|Lx\|_2^2$ is just the Tikhonov regularization term which comes from the prior distribution of x that encodes the structure we expect to enforce on x.

In this paper, we consider iterative methods for solving the two equivalent regularized inverse problems

$$\min_{x\in\mathbb{R}^n}\{\|Ax-b\|_2^2+\lambda x^TMx\}\quad\text{or}\quad\min_{x\in\mathbb{R}^n}\{\|Ax-b\|_2^2+\lambda\|Lx\|_2^2\} \tag{1.4}$$

within the subspace projection regularization framework

$$\min_{x\in\mathcal{X}_k}x^TMx,\quad \mathcal{X}_k=\{x:\min_{x\in\mathcal{S}_k}\|Ax-b\|_2\} \tag{1.5}$$

to avoid choosing in advance regularization parameters. A series of solution subspaces $\mathcal{S}_k\subseteq\mathbb{R}^n$ of dimension $k=1,2,\dots$ should be constructed to incorporate prior properties of the solution encoded by the regularizer x^TMx. For standard-form regularization with $M=I$, the most popular iterative regularization method is LSQR [44] with an early stopping rule, which projects (1.1) onto a sequence of lower dimensional Krylov subspaces to approximate x_{true} [5, 43]. The iteration should stop early to overcome semiconvergence by some criteria such as L-curve, discrepancy principle, or GCV [20, 24, 42]. For general-form regularization that $M\neq I$, if L is already available or the decomposition $M=L^TL$ can be obtained without too much computation, there are many methods that deal with the regularizer $\|Lx\|_2^2$ instead of x^TMx. For an invertible L, we can use L as a preconditioner by the substitution $y=Lx$; see [7, 8, 9] for preconditioned methods for Bayesian inverse problems. Otherwise, (1.4) can be transformed to the standard form by using the A-weighted pseudoinverse $L_A^\dagger$; see [14] or [25, section 2.3] for details. However, such transformations are often computationally unfeasible for large-scale matrices. For large-scale A and L, there are some other iterative regularization methods, such as the modified truncated SVD method [2, 31], joint bidiagonalization method [33, 34, 36, 41], methods based on randomized generalized SVD (GSVD) of $\{A, L\}$ [55, 57, 58], methods based on generalized Krylov subspace [32, 39, 46], and so on.

However, all of the above methods require L, which is not available in some scenarios of applications. For example, for the Matérn class of covariance functions describing the prior of x, which are also called Matérn kernels [18, 49], the corresponding M can be large scale and dense and thus computing L is extremely expensive. Another class of frequently encountered examples arises in the lagged diffusivity fixed point (LDFP) iteration method for nonlinear regularizers [11, 56], such as total variation [50] or the Perona–Malik [45] regularizer used in image reconstruction and electrical impedance tomography, where at each outer iteration of LDFP, a large-scale M is constructed to linearize the nonlinear regularizer and a corresponding regularization problem (1.4) needs to be solved. For these cases, we have to deal with x^TMx instead of $\|Lx\|_2^2$. If M is positive definite and M^{-1} (i.e., the covariance matrix of the prior) is already known, which is often the case for Matérn kernels, the generalized Golub–Kahan bidiagonalization method [13] is very efficient. For many cases that M^{-1} is unknown, a preconditioned LSQR method called MLSQR has been proposed where a linear system $Mx = y$ needs to be iteratively solved at each iteration [1], and it has been used in the inner iteration of LDFP for many applications; see e.g. [4, 22, 29]. To successfully apply MLSQR, a big challenge is that M is often noninvertible or nearly singular. Even if M is invertible, it is often the case that M has a very large condition number, which results in too many iterations being needed for solving $Mx = y$, thus significantly reducing the efficiency of this algorithm. In some work such as [10, 23], the researchers suggest replacing M by $M_\delta = M + \delta I$ to make M_δ positive definite and well-conditioned. However, the proper value of δ can only be set by numerical trials, and there may be an accuracy sacrifice of the solution since the target problem has been changed.

In this paper, we propose a new Krylov subspace based regularization method to deal with the regularizer x^TMx for positive semidefinite matrix M, and our method does not need to replace M by a positive definite M_δ. To this end, we first design an iterative process that generates a series of vectors spanning the solution subspaces, where a proper preconditioner is implicitly constructed and exploited. This preconditioner is proper in the sense that the generated solution subspaces can incorporate prior properties of the desired solution encoded by the regularizer x^TMx, resulting in an iterative regularized solution of high quality. The main contributions of this paper are listed as follows:

- We design an iterative process similar to the Golub–Kahan bidiagonalization (GKB) to generate a series of solution subspaces. This process is proven to be mathematically equivalent to the standard GKB process of a preconditioned A, where a right preconditioner is implicitly used, thereby we name it the preconditioned GKB (pGKB) process.
- By giving explicit expression of the solution subspace using the GSVD of $\{A, L\}$, we show that the above solution subspace can incorporate prior properties of the desired solution encoded by the regularizer. Based on the pGKB process, we propose a subspace projection regularization algorithm via projecting the original problem onto the solution subspace at each iteration.

- We analyze the regularization property of the above proposed algorithm. We prove that the iterative solution has a filtered GSVD expansion form, where some dominant GSVD components are captured and others are filtered out. This reveals that the algorithm exhibits typical semiconvergence behavior, where the iteration number k plays the role of regularization parameter.
- To overcome instabilities arising from semiconvergence, two pGKB based hybrid regularization algorithms are proposed, where Tikhonov regularization is applied to the projected small-scale problem at each iteration. To efficiently determine regularization parameters for the projected small-scale problems, the weighted GCV (WGCV) method and the "secant update" method based on the discrepancy principle are adopted, respectively.

All these proposed iterative algorithms are tested on both small-scale and large-scale linear inverse problems to show excellent effectiveness and performance.

The paper is organized as follows. In section 2, we review basic properties of the general-form Tikhonov regularization using the GSVD of $\{A, L\}$. In section 3, we design the pGKB process and propose the pGKB based subspace projection regularization (pGKB_SPR) algorithm. In section 4, we analyze the regularization effect of pGKB_SPR and reveal the semiconvergence behavior of it. To overcome instabilities caused by semiconvergence, in section 5 we propose two pGKB based hybrid regularization algorithms. In section 6, we choose several small-scale and large-scale linear inverse problems to test the proposed algorithms. Finally, we give some concluding remarks in section 7.

Throughout the paper, we denote by I and $\mathbf{0}$ the identity matrix and zero matrix/vector, respectively, with orders clear from the context, and denote by $\mathrm{span}\{\cdot\}$ the subspace spanned by a group of vectors or columns of a matrix.

2. General-form Tikhonov regularization and GSVD.

Although in this paper, the square root decomposition $M = L^T L$ is not needed, it is convenient to use L to make some analysis. Since M is positive semidefinite, we write L in the compact form, i.e., $L \in \mathbb{R}^{p\times n}$ with $p \le n$ and $\mathrm{rank}(L) = p$. Taking the gradient of x in (1.4) leads to

$$(A^T A + \lambda M)x = A^T b. \tag{2.1}$$

Denote by $\mathcal{N}(\cdot)$ the null space of a matrix. In order to ensure that there exists a unique regularized solution for any $\lambda > 0$, the sufficient and necessary condition is

$$\mathrm{rank}(A^T A + \lambda M) = n \iff \mathcal{N}(A) \cap \mathcal{N}(M) = \{\mathbf{0}\}, \tag{2.2}$$

and the solution is

$$x_\lambda = (A^T A + \lambda M)^{-1} A^T b. \tag{2.3}$$

Note that $\mathcal{N}(A) \cap \mathcal{N}(M) = \{\mathbf{0}\} \Leftrightarrow \mathcal{N}(A) \cap \mathcal{N}(L) = \{\mathbf{0}\}$ since $\mathcal{N}(L) = \mathcal{N}(M)$. The above solution can be expressed in a more convenient form using the GSVD of $\{A, L\}$ [54], which is given by

$$A = U_A D_A Z^{-1}, \quad L = U_L D_L Z^{-1}, \tag{2.4}$$

where $U_A \in \mathbb{R}^{m\times m}$ and $U_L \in \mathbb{R}^{p\times p}$ are orthogonal, $Z \in \mathbb{R}^{n\times n}$ is invertible, and

$$D_A = \begin{array}{cc} \left(\begin{array}{ccc} I & & \\ & \Sigma_A & \\ & & \mathbf{0} \end{array}\right) & \begin{array}{c} r \\ q \\ m-r-q \end{array} \\ \begin{array}{ccc} r & q & n-r-q \end{array} & \end{array}, \quad D_L = \begin{array}{cc} \left(\begin{array}{ccc} \mathbf{0} & & \\ & \Sigma_L & \\ & & I \end{array}\right) & \begin{array}{c} p+r-n \\ q \\ n-r-q \end{array} \\ \begin{array}{ccc} r & q & n-r-q \end{array} & \end{array}$$

with diagonal matrices $\Sigma_A = \mathrm{diag}(\sigma_{r+1}, \ldots, \sigma_{r+q})$ and $\Sigma_L = \mathrm{diag}(\rho_{r+1}, \ldots, \rho_{r+q})$. The identity $\Sigma_A^T \Sigma_A + \Sigma_L^T \Sigma_L = I$ holds. If we arrange σ_i and ρ_i in decreasing order of values such that

$$1 > \sigma_{r+1} \geq \sigma_{r+2} \geq \cdots \geq \sigma_{r+q} > 0, \quad 0 < \rho_{r+1} \leq \rho_{r+2} \leq \cdots \leq \rho_{r+q} < 1, \tag{2.5}$$

then $\sigma_i^2 + \rho_i^2 = 1$, and $\gamma_i := \sigma_i / \rho_i$ is called the ith generalized singular value of $\{A, L\}$. We define $\gamma_i = \infty$ for $1 \leq i \leq r$ and $\gamma_i = 0$ for $r + q + 1 \leq i \leq n$. Note that $\dim(\mathcal{N}(A)) = n - r - q$ and $\dim(\mathcal{N}(L)) = r$, thereby $q = n - \dim(\mathcal{N}(A) \oplus \mathcal{N}(L))$.

Denote the columns of U_A and Z by $\{u_{A,i}\}_{i=1}^m$ and $\{z_i\}_{i=1}^n$, respectively. The discrete Picard condition (DPC) plays a central role in the regularization of discrete ill-posed problems, which says that the Fourier coefficients $|u_{A,i}^T b_{\mathrm{true}}|$ on average decay to zero faster than the corresponding γ_i, where $b_{\mathrm{true}} = A x_{\mathrm{true}}$. Any regularization is based on an underlying requirement that the DPC is satisfied, only under which can one compute a regularized solution with some accuracy [25, section 4.5]. Using the decomposition (2.4), the Tikhonov solution x_λ can be written as

$$x_\lambda = \sum_{i=1}^{r} (u_{A,i}^T b) z_i + \sum_{i=r+1}^{r+q} \frac{\gamma_i^2}{\gamma_i^2 + \lambda} \frac{u_{A,i}^T b}{\sigma_i} z_i. \tag{2.6}$$

The factors $f_i := \gamma_i^2 / (\gamma_i^2 + \lambda)$ can be viewed as filters applied to noisy coefficients $u_{A,i}^T b$. A proper value of λ should satisfy that $f_i \approx 1$ for small i and $f_i \approx 0$ for large i, and thereby dampen noisy components appearing in the regularized solution. Although there are several methods for choosing regularization parameter, one big challenge is that in order to find a suitable λ, many different values of λ must be tried to solve (1.4) in advance or the GSVD of $\{A, L\}$ should be computed, which is computationally expensive for large-scale problems.

For large-scale problems, an alternative is the subspace projection regularization method [15, section 3.3], which seeks to compute a series of x_k as the solution to

$$\min_{x \in \mathcal{X}_k} x^T M x, \quad \mathcal{X}_k = \{x : \min_{x \in \mathcal{S}_k} \|Ax - b\|_2\}, \tag{2.7}$$

where $\mathcal{S}_k$ is the subspace of $\mathbb{R}^n$ of dimension $k = 1, 2, \ldots$ and the iteration proceeds until an early stopping criterion is satisfied to overcome underregularizing caused by semiconvergence.

Remark 2.1. Under the condition (2.2), the subspace projected problem (2.7) has a unique solution for $k = 1, \ldots, n$. Let S_k be an $n \times k$ column orthonormal matrix whose columns span $\mathcal{S}_k$. By writing any $x \in \mathcal{S}_k$ as $x = S_k y$, we get the solution to (2.7) as $x_k = S_k y_k$, where y_k is the solution to $\min_{x \in \mathcal{Y}_k} \|L S_k y\|_2$, $\mathcal{Y}_k = \{y : \min_{y \in \mathbb{R}^k} \|A S_k y - b\|_2\}$. By [14, Theorem 2.1], there exists a unique solution y_k if and only if $\mathcal{N}(A S_k) \cap \mathcal{N}(L S_k) = \{\mathbf{0}\}$, which is equivalent to $\mathrm{rank}((\begin{smallmatrix} A \\ L \end{smallmatrix}) S_k) = n$. Since S_k has full column rank, it means that $\mathrm{rank}((\begin{smallmatrix} A \\ L \end{smallmatrix})) = n \Leftrightarrow \mathcal{N}(A) \cap \mathcal{N}(L) = \{\mathbf{0}\}$. Thus (2.2) is the sufficient and necessary condition for (2.7) having a unique solution.

We call $\mathcal{S}_k$ the *solution subspace,* which should be constructed elaborately so that the prior information about the desired solution is incorporated in $\mathcal{S}_k$. For the general-form regularizer, from the expression (2.6) we know that the most ideal choice is $\mathcal{S}_k = \mathrm{span}\{Z_k\}$, where $Z_k = (z_1, \ldots, z_k)$. This leads to the truncated GSVD (TGSVD) solution

$$x_k^{\mathrm{TGSVD}} = \sum_{i=1}^{k} \frac{u_{A,i}^T b}{\sigma_i} z_i, \tag{2.8}$$

where we let $\sigma_i = 1$ for $1 \leq i \leq r$ [25, section 3.2]. By truncating the GSVD at a proper k, the obtained solution can capture the main information corresponding to dominant right generalized singular vectors while suppressing noise corresponding to others. For large-scale problems that the GSVD is computationally expensive, a legitimate subspace projection regularization method should construct a series of solution subspaces that can approximate well the dominant right generalized singular vectors z_i as the iterations progress. To this end, we first gain an insight into the relation between z_i and the matrix pair $\{A, M\}$.

Proposition 2.1. *Let $G = A^TA + \alpha M$ with any $\alpha > 0$. The generalized eigenvalues of $A^TAz = \xi Gz$ in decreasing order are*

$$\underbrace{1, \ldots, 1}_{r}, \underbrace{\gamma_{r+1}^2/(\gamma_{r+1}^2 + \alpha), \ldots, \gamma_{r+q}^2/(\gamma_{r+q}^2 + \alpha)}_{q}, \underbrace{0, \ldots, 0}_{n-r-q}, \tag{2.9}$$

and the corresponding generalized eigenvectors are $\{z_i\}_{i=1}^n$.

Proof. Observing from (2.4) that

$$A^TA = Z^{-T}D_A^TD_AZ^{-1}, \quad M = L^TL = Z^{-T}D_L^TD_LZ^{-1},$$

where $D_A^TD_A$ and $D_L^TD_L$ are diagonal matrices of the following form:

$$D_A^TD_A = \mathrm{diag}(\underbrace{1, \ldots, 1}_{r}, \underbrace{\sigma_{r+1}^2, \ldots, \sigma_{r+q}^2}_{q}, \underbrace{0, \ldots, 0}_{n-r-q}),$$
$$D_L^TD_L = \mathrm{diag}(\underbrace{0, \ldots, 0}_{r}, \underbrace{\rho_{r+1}^2, \ldots, \rho_{r+q}^2}_{q}, \underbrace{1, \ldots, 1}_{n-r-q}),$$

we have $G = A^TA + \alpha M = Z^{-T}D_\alpha Z^{-1}$ with

$$D_\alpha = \mathrm{diag}(\underbrace{1, \ldots, 1}_{r}, \underbrace{\sigma_{r+1}^2 + \alpha\rho_{r+1}^2, \ldots, \sigma_{r+q}^2 + \alpha\rho_{r+q}^2}_{q}, \underbrace{\alpha, \ldots, \alpha}_{n-r-q}). \tag{2.10}$$

Therefore, the $\{z_i\}_{i=1}^n$ are generalized eigenvectors of $A^TAz = \xi Gz$ with generalized eigenvalues being the diagonals of $D_A^TD_AD_\alpha^{-1}$, which are

$$\begin{aligned}&\underbrace{1, \ldots, 1}_{r}, \underbrace{\sigma_{r+1}^2/(\sigma_{r+1}^2 + \alpha\rho_{r+1}^2), \ldots, \sigma_{r+q}^2/(\sigma_{r+q}^2 + \alpha\rho_{r+q}^2)}_{q}, \underbrace{0, \ldots, 0}_{n-r-q}\\ &= \underbrace{1, \ldots, 1}_{r}, \underbrace{\gamma_{r+1}^2/(\gamma_{r+1}^2 + \alpha), \ldots, \gamma_{r+q}^2/(\gamma_{r+q}^2 + \alpha)}_{q}, \underbrace{0, \ldots, 0}_{n-r-q}.\end{aligned}$$

Note that $\gamma_i^2/(\gamma_i^2 + \alpha)$ increases with respect to γ_i. The desired result is obtained. ☐

We remark that α can be any positive number; more discussions about setting a good value of α can be found at the end of section 3. Note that G is positive definite. Inspired by Proposition 2.1, we consider constructing solution spaces in the G-inner product space, where the G-inner product in $\mathbb{R}^n$ is defined by $\langle x, x'\rangle_G = x^TGx'$ for any $x, x' \in \mathbb{R}^n$. We hope those dominant z_i can be well captured by the solution subspaces so that the iterative solution will incorporate main features described by those z_i.

3. pGKB and subspace projection regularization algorithm.

For standard-form regularization, the most popular LSQR algorithm is based upon the GKB that constructs solution subspaces in the standard inner product space $\mathbb{R}^n$. In order to construct proper solution subspaces for regularizer x^TMx, we consider the

GKB process using the G-inner product. Note that the standard GKB process needs A and its transpose A^T. For the G-inner product case, we need the matrix-form expression of the adjoint of a linear operator between the G- and standard inner product Hilbert spaces $(\mathbb{R}^n, \langle\cdot,\cdot\rangle_G)$ and $(\mathbb{R}^m, \langle\cdot,\cdot\rangle_2)$.

LEMMA 3.1. *For the linear operator $A : (\mathbb{R}^n, \langle\cdot,\cdot\rangle_G) \to (\mathbb{R}^m, \langle\cdot,\cdot\rangle_2)$ between the two Hilbert spaces, define $A^{*_G} : (\mathbb{R}^m, \langle\cdot,\cdot\rangle_2) \to (\mathbb{R}^n, \langle\cdot,\cdot\rangle_G)$, which is the adjoint of A, by $\langle Ax, y\rangle_2 = \langle x, A^{*_G}y\rangle_G$ for any $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$. Then the matrix form of A^{*_G} is*

$$A^{*_G} = G^{-1}A^T. \tag{3.1}$$

Proof. First note that A^{*_G} is well-defined, which can be found in any functional analysis textbook. Since $\langle Ax, y\rangle_2 = x^TA^Ty$ and $\langle x, A^{*_G}y\rangle_G = x^TGA^{*_G}y$, letting $x^TA^Ty = x^TGA^{*_G}y$ for any vectors x and y, it must hold that $A^T = GA^{*_G}$, and we immediately obtain (3.1). □

For the standard inner product in $\mathbb{R}^n$, i.e., $G = I$, we have $A^{*_G} = A^T$, which is just the standard matrix transpose. Now we seek to construct solution subspaces $\mathcal{S}_k \subseteq (\mathbb{R}^n, \langle\cdot,\cdot\rangle_G)$ to iteratively solve $\min_{x\in\mathcal{S}_k} \|Ax - b\|_2$. With the help of Lemma 3.1, the construction of $\mathcal{S}_k$ can be done by the GKB process applied to A and b between the two Hilbert spaces $(\mathbb{R}^n, \langle\cdot,\cdot\rangle_G)$ and $(\mathbb{R}^m, \langle\cdot,\cdot\rangle_2)$. This process can be written as the following recursive relations:

$$\beta_1u_1 = b, \quad \alpha_1w_1 = A^{*_G}u_1, \tag{3.2}$$

$$\beta_{i+1}u_{i+1} = Aw_i - \alpha_iu_i, \tag{3.3}$$

$$\alpha_{i+1}w_{i+1} = A^{*_G}u_{i+1} - \beta_{i+1}w_i, \tag{3.4}$$

where $u_i \in (\mathbb{R}^m, \langle\cdot,\cdot\rangle_2)$, $w_i \in (\mathbb{R}^n, \langle\cdot,\cdot\rangle_G)$, and α_i, β_i should be computed such that $\|u_i\|_2 = \|w_i\|_G = 1$. Thus we have $u_1 = b/\beta_1$ with $\beta_1 = \|b\|_2$. Using the matrix-form expression of A^{*_G} we have

$$\alpha_{i+1}w_{i+1} = G^{-1}A^Tu_{i+1} - \beta_{i+1}w_i \tag{3.5}$$

with $\alpha_{i+1} = \|G^{-1}A^Tu_{i+1} - \beta_{i+1}w_i\|_G$. Note that for $i = 0$ we define $w_0 = \mathbf{0}$.

Based on the coupled recursive relations (3.3) and (3.5), we get the pGKB process, which is summarized in Algorithm 3.1. The usage of the name "preconditioned" will be explained later.

After k steps, the pGKB generates two groups of vectors $\{u_i\}_{i=1}^{k+1}$ and $\{w_i\}_{i=1}^{k+1}$. Define two matrices as $U_k = (u_1, \ldots, u_k)$ and $W_k = (w_1, \ldots, w_k)$. Then by (3.2), (3.3), and (3.5), we have the matrix-form recursive relations

$$\beta_1U_{k+1}e_1 = b, \tag{3.6}$$

$$AW_k = U_{k+1}B_k, \tag{3.7}$$

$$G^{-1}A^TU_{k+1} = W_kB_k^T + \alpha_{k+1}w_{k+1}e_{k+1}^T, \tag{3.8}$$

where

$$B_k = \begin{pmatrix} \alpha_1 & & & \\ \beta_2 & \alpha_2 & & \\ & \beta_3 & \ddots & \\ & & \ddots & \alpha_k \\ & & & \beta_{k+1} \end{pmatrix} \in \mathbb{R}^{(k+1)\times k} \tag{3.9}$$

Algorithm 3.1. pGKB.

```
Input: $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $M \in \mathbb{R}^{n\times n}$, $\alpha > 0$
$G = A^TA + \alpha M$
$\beta_1 = \|b\|_2$, $u_1 = b/\beta_1$
Compute $s$ by solving $Gs = A^Tu_1$
$\alpha_1 = (s^TGs)^{1/2}$, $w_1 = s/\alpha_1$
for $i = 1, 2, \ldots, k$, do
    $r = Aw_i - \alpha_i u_i$
    $\beta_{i+1} = \|r\|_2$, $u_{i+1} = r/\beta_{i+1}$
    Compute $s$ by solving $Gs = A^Tu_{i+1}$
    $s = s - \beta_{i+1}w_i$
    $\alpha_{i+1} = (s^TGs)^{1/2}$, $w_{i+1} = s/\alpha_{i+1}$
end for
Output: $\{\alpha_i, \beta_i\}_{i+1}^{k+1}$, $\{u_i, w_i\}_{i+1}^{k+1}$
```

and e_1 and e_{k+1} are the first and $(k+1)$th columns of the identity matrix of order $k+1$, respectively. In fact, U_{k+1} is a 2-orthonormal matrix and W_k is a G-orthonormal matrix, which will be proved in the following lemma. Thus by (3.7) we have $B_k = U_{k+1}^TAW_k$, which implies that B_k is a projection of A onto two subspaces, span$\{U_{k+1}\}$ and span$\{W_k\}$. Now we analyze the structure of these two subspaces.

LEMMA 3.2. *The group of vectors $\{u_i\}_{i=1}^k$ is a 2-orthonormal basis of the Krylov subspace*

$$\mathcal{K}_k(AG^{-1}A^T, b) = \mathrm{span}\{(AG^{-1}A^T)^ib\}_{i=0}^{k-1}, \tag{3.10}$$

and $\{w_i\}_{i=1}^k$ is a G-orthonormal basis of the Krylov subspace

$$\mathcal{K}_k(G^{-1}A^TA, G^{-1}A^Tb) = \mathrm{span}\{(G^{-1}A^TA)^iG^{-1}A^Tb\}_{i=0}^{k-1}. \tag{3.11}$$

Proof. In order to get more insights into the pGKB process, we give two proofs.

The first proof. This proof exploits the property of the GKB process of A between the two Hilbert spaces $(\mathbb{R}^n, \langle\cdot,\cdot\rangle_G)$ and $(\mathbb{R}^m, \langle\cdot,\cdot\rangle_2)$, which states that $\{u_i\}_{i=1}^k$ and $\{z_i\}_{i=1}^k$ are the 2-orthonormal basis and G-orthonormal basis of the Krylov subspaces $\mathcal{K}_k(AA^{*_G}, b)$ and $\mathcal{K}_k(A^{*_G}A, A^{*_G}b)$, respectively. Note that $AA^{*_G} = AG^{-1}A^T$, $A^{*_G}A = G^{-1}A^TA$, and $A^{*_G}b = G^{-1}A^Tb$. The proof is completed.

The second proof. Suppose the Cholesky factorization of G is $G = R^TR$. Let $\bar{A} = AR^{-1}$, $v_i = Rw_i$, and $V_k = (v_1, \ldots, v_k)$. Then by (3.7) and (3.8) we have

$$\bar{A}V_k = U_{k+1}B_k, \quad \bar{A}^TU_{k+1} = V_kB_k^T + \alpha_{k+1}v_{k+1}e_{k+1}^T. \tag{3.12}$$

Since $\|w_i\|_G = 1$, we have $\|v_i\|_2 = (w_iR^TRw_i)^{1/2} = (w_i^TGw_i)^{1/2} = 1$. Combining relations (3.6) and (3.12) and using $\|u_i\|_2 = \|v_i\|_2$, we know that $\{u_i\}_{i=1}^k$ and $\{v_i\}_{i=1}^k$ are the Lanczos vectors generated by the GKB process of $\bar{A}$ with starting vector b under the standard inner product. Therefore, $\{u_i\}_{i=1}^k$ and $\{v_i\}_{i=1}^k$ are 2-orthonormal bases of Krylov subspaces $\mathcal{K}_k(\bar{A}\bar{A}^T, b) = \mathcal{K}_k(AG^{-1}A^T, b)$ and $\mathcal{K}_k(\bar{A}^T\bar{A}, \bar{A}^Tb) = \mathcal{K}_k(R^{-T}A^TAR^{-1}, R^{-T}A^Tb)$, respectively. Using $w_i = R^{-1}v_i$, we obtain that $\{w_i\}_{i=1}^k$ is a G-orthonormal basis of the subspace

$$R^{-1}\mathcal{K}_k(\bar{A}^T\bar{A}, \bar{A}^Tb) = R^{-1}\mathrm{span}\{(R^{-T}A^TAR^{-1})^iR^{-T}A^Tb\}_{i=0}^{k-1}.$$

Note that

$$\begin{aligned} R^{-1}(R^{-T}A^TAR^{-1})^iR^{-T}A^Tb &= R^{-1}R^{-T}A^T(AR^{-1}R^{-T}A^T)^ib \\ &= G^{-1}A^T(AG^{-1}A^T)^ib \\ &= (G^{-1}A^TA)^iG^{-1}A^Tb. \end{aligned}$$

The proof is completed. □

The second proof explains why we use the name "preconditioned" for Algorithm 3.1. The pGKB is essentially the standard GKB process of the preconditioned matrix AR^{-1} with starting vector b, where R^{-1} is the right preconditioner. It reduces AR^{-1} to a bidiagonal form B_k while generating two orthonormal matrices U_{k+1} and V_{k+1}, and the desired vectors w_i are obtained by $w_i = R^{-1}v_i$. In Algorithm 3.1, the Cholesky factor R and R^{-1} need not be explicitly computed to get w_i, while instead a linear system $Gs = A^Tu_i$ must be solved.

Based on the pGKB process, we propose the subspace projection regularization algorithm by letting $\mathcal{S}_k = \text{span}\{W_k\}$ and solving (2.7). Suppose the k-step pGKB does not terminate, that is, $\alpha_i, \beta_i \neq 0$ for $1 \leq i \leq k$. Then B_k has full column rank and, thus,

$$\begin{aligned} \min_{x=W_ky} \|Ax-b\|_2 &= \min_{y\in\mathbb{R}^k} \|AW_ky - U_{k+1}\beta_1e_1\|_2 \\ &= \min_{y\in\mathbb{R}^k} \|U_{k+1}B_ky - U_{k+1}\beta_1e_1\|_2 \\ &= \min_{y\in\mathbb{R}^k} \|B_ky - \beta_1e_1\|_2, \end{aligned}$$

which has a unique solution. By writting $x = W_ky$ with $y \in \mathbb{R}^k$ for any $x \in \mathcal{S}_k$, the problem (2.7) with $\mathcal{S}_k = \text{span}\{W_k\}$ becomes

$$\min_{y\in\mathcal{Y}_k} yW_k^TMW_ky, \quad \mathcal{Y}_k = \left\{ y : \min_{y\in\mathbb{R}^k} \|B_ky - \beta_1e_1\|_2 \right\}.$$

Since $\mathcal{Y}_k$ has only one element, the solution to (2.7) is

$$x_k = W_ky_k, \quad y_k = \arg\min_{y\in\mathbb{R}^k} \|B_ky - \beta_1e_1\|_2 \tag{3.13}$$

for $k = 1, 2 \ldots$. This is a very similar procedure to the LSQR solver for least squares problems. In practical computations, there is a recursive formula to update x_{k+1} from x_k without solving the projected problem $\min_y \|B_ky - \beta_1e_1\|_2$ at each iteration. This will be discussed later. Now we investigate the structure of solution subspace $\mathcal{S}_k$.

PROPOSITION 3.1. *Let the solution subspace be* $\mathcal{S}_k = \text{span}\{W_k\}$. *Then*

$$\mathcal{S}_k = \text{span}\{Z(D_\alpha^{-1}D_A^TD_A)^iD_\alpha^{-1}D_A^TU_A^Tb\}_{i=0}^{k-1}. \tag{3.14}$$

Proof. By Lemma 3.2 we have $\mathcal{S}_k = \text{span}\{(G^{-1}A^TA)^iG^{-1}A^Tb\}_{i=0}^{k-1}$. Using the GSVD expression (2.4) and Proposition 2.1, we have

$$\begin{aligned} G^{-1}A^Tb &= ZD_\alpha^{-1}Z^TZ^{-T}D_A^TU_A^Tb = ZD_\alpha^{-1}D_A^TU_A^Tb, \\ G^{-1}A^TA &= ZD_\alpha^{-1}Z^TZ^{-T}D_A^TD_AZ^{-1} = ZD_\alpha^{-1}D_A^TD_AZ^{-1}. \end{aligned}$$

Therefore, we obtain

$$\begin{aligned} (G^{-1}A^TA)^iG^{-1}A^Tb &= (ZD_\alpha^{-1}D_A^TD_AZ^{-1})^iZD_\alpha^{-1}D_A^TU_A^Tb \\ &= Z(D_\alpha^{-1}D_A^TD_A)^iZ^{-1}ZD_\alpha^{-1}D_A^TU_A^Tb \\ &= Z(D_\alpha^{-1}D_A^TD_A)^iD_\alpha^{-1}D_A^TU_A^Tb \end{aligned}$$

for $i = 0, 1, \ldots, k-1$, which is the desired result. □

Algorithm 3.2. Updating procedure.

```
Let $x_0 = \mathbf{0},\ p_1 = w_1,\ \bar{\phi}_1 = \beta_1,\ \bar{\rho}_1 = \alpha_1$
for $i = 1, 2, \ldots, k$, do
    $\rho_i = (\bar{\rho}_i^2 + \beta_{i+1}^2)^{1/2}$
    $c_i = \bar{\rho}_i/\rho_i,\ s_i = \beta_{i+1}/\rho_i$
    $\theta_{i+1} = s_i\alpha_{i+1},\ \bar{\rho}_{i+1} = -c_i\alpha_{i+1}$
    $\phi_i = c_i\bar{\phi}_i,\ \bar{\phi}_{i+1} = s_i\bar{\phi}_i$
    $x_i = x_{i-1} + (\phi_i/\rho_i)p_i$
    $p_{i+1} = w_{i+1} - (\theta_{i+1}/\rho_i)p_i$
end for
```

As will be shown in section 4, the subspace $\mathcal{S}_k$ can be used to approximate dominant vectors among $\{z_i\}_{i=1}^n$. Hence we can expect that the iterative regularized solution can include prior properties about the desired solution encoded by the regularizer x^TMx. We emphasize that the good property of $\mathcal{S}_k$ stems from implicitly utilizing the appropriate preconditioner R for constructing W_k. Here the term preconditioned is not aimed at accelerating the convergence of iterative solvers but rather at forcing some specific regularity into the associated solution subspace. In [21], the authors proposed a preconditioned conjugate gradient algorithm for discrete linear inverse problems with block Toeplitz coefficient matrices, which shares a similar spirit. In fact, the regularized solution x_k has a filtered GSVD expansion form, which sheds light on the regularization effect of the proposed algorithm. We will analyze it in the next section.

Now we discuss how to efficiently update solutions and stop the iteration early to get a good final regularized solution. By applying Givens QR factorization to (3.13), the updating procedure for x_k can be derived similarly to that for LSQR; see [44] for details. Starting from $x_0 = \mathbf{0}$, x_i is computed recursively by the following algorithm.

The identity

$$\bar{\phi}_{k+1} = \|B_ky_k - \beta_1e_1\|_2 = \|Ax_k - b\|_2 \tag{3.15}$$

holds, similarly to that for LSQR [44]. Thus the residual norm can be updated very quickly by Algorithm 3.2 without explicitly computing $\|Ax_k - b\|_2$. Note that $\bar{\phi}_{k+1}$ decreases monotonically since x_k minimizes $\|Ax - b\|_2$ in the gradually expanding subspace span$\{W_k\}$.

In order to estimate the optimal early stopping iteration, if we have an accurate estimate of $\|e\|$, the discrepancy principle (DP) is a common choice. It states that we should stop iteration at the first k satisfying

$$\bar{\phi}_{k+1} = \|Ax_k - b\|_2 \leq \tau\|e\|_2, \tag{3.16}$$

where $\tau > 1$ slightly. The DP method usually suffers from underestimating the optimal k and thus the solution is overregularized. Another approach is the L-curve criterion, which does not need $\|e\|$ in advance. We plot the L-curve

$$\left(\log\|Ax_k - b\|_2, \log(x_k^TMx_k)^{1/2}\right) = \left(\log\bar{\phi}_{k+1}, \log(x_k^TMx_k)^{1/2}\right) \tag{3.17}$$

and choose the k corresponding to the corner of it as a good estimate of the optimal early stopping iteration. We remark that an underlying assumption for the use of the

Algorithm 3.3. pGKB_SPR.

```
Input: $A$, $M$, $\alpha > 0$, $b$, $x_0 = \mathbf{0}$          ▷ require $\tau\|e\|_2$ for DP
for $k = 1, 2, \ldots,$ do
    Compute $u_k$, $w_k$, $\alpha_k$, $\beta_k$ by pGKB
    Compute $\rho_k$, $\theta_{k+1}$, $\bar{\rho}_{k+1}$, $\phi_k$, $\bar{\phi}_{k+1}$ by updating procedure
    Compute $x_k$, $p_{k+1}$ by updating procedure
    if Early stopping criterion is satisfied then          ▷ DP or L-curve
        Denote the estimated iteration by $k_1$, terminate iteration at $k_1$
    end if
end for
Output: Final regularized solution $x_{k_1}$
```

L-curve criterion is that $\|Ax_k - b\|_2$ and $(x_k^T M x_k)^{1/2}$ are monotonically decreasing and increasing with respect to k, respectively. Although the latter cannot be proved rigorously, we find experimentally that the L-curve criterion almost always works well.

The whole process of the pGKB_SPR algorithm is summarized in Algorithm 3.3.

At the end of this section, we discuss several implementation details for the pGKB process, which are important for increasing computational efficiency, especially for large-scale problems. There are three main issues that need to be considered.

- The pGKB process has the structure of nested inner-out iteration, where at each outer iteration, a linear system $Gs = A^T u_i$ needs to be solved. Although for some cases where A and M have special structures, this system can be solved via a direct matrix factorization, for most large-scale problems, however, the only advisable approach is using an iterative solver, such as the conjugate gradient (CG) or minimum residual algorithm. Meanwhile, The value of $\alpha > 0$ should be set such that the condition number of $G = A^T A + \alpha M$ is small to make the iterative solver converge quickly. The default value $\alpha = 1$ is often fruitful, and we also try other moderate values $\alpha \in [0.001, 100]$ in the numerical experiments.
- For large-scale problems, iteratively solving $Gs = A^T u_i$ may still be costly, especially when the solution accuracy is high. Since there is a limit to the accuracy of the best regularized solution, which is affected by the noise level and other factors, we can expect that the accuracy of the final regularized solution will not be affected even if inner iterations are not computed exactly. When the noise level $\varepsilon = \|e\|_2 / \|b_{\mathrm{true}}\|_2$ is not extremely small, we numerically find that the solution accuracy of $Gs = A^T u_i$ can be relaxed considerably (the stopping tolerance for the MATLAB function `pcg.m` can be set larger than the default value `1e-6`), which improves the overall efficiency of the algorithm.
- Due to the computational errors and rounding errors coming from inaccurate inner iterations and finite precision arithmetic, respectively, the 2- and G-orthogonality of computed u_i and w_i gradually lose, which leads to a delay of convergence of iterative solutions. In our implementation, we do full reorthogonalization on u_i and w_i based on the modified Gram–Schmidt orthogonalization to maintain their numerical 2- and G-orthogonality to ensure the normal convergence behavior.

In this paper, the above issues are only investigated numerically. We will make a theoretical analysis of the required accuracy of inner iterations as well as efficient reorthogonalization strategies in the forthcoming work.

4. Regularization property of pGKB_SPR.

The dominant GSVD components of $\{A, L\}$ can be approximated well by the solution subspaces generated by pGKB as the iteration proceeds. This is a desirable property for a regularization method to ensure that the obtained solution captures the main information corresponding to dominant right generalized singular vectors while filtering out noise corresponding to others.

LEMMA 4.1. *Suppose the singular values of B_k are $\theta_1^{(k)} > \theta_2^{(k)} > \cdots > \theta_k^{(k)} > 0$. Then $(\theta_i^{(k)})^2$ converges to one of the generalized eigenvalues of $A^TAz = \xi Gz$ as k increases.*

Proof. Since $\alpha_1 w_1 = G^{-1}A^Tu_1$ and $\beta_1 u_1 = b$, we have

$$\alpha_1\beta_1 w_1 = G^{-1}A^Tb. \tag{4.1}$$

By (3.7) and (3.8), we have

$$\begin{aligned} A^TAW_k &= A^TU_{k+1}B_k \\ &= (GW_kB_k^T + \alpha_{k+1}Gw_{k+1}e_{k+1}^T)B_k \\ &= GW_k(B_k^TB_k) + \alpha_{k+1}\beta_{k+1}Gw_{k+1}e_{k+1}^T. \end{aligned} \tag{4.2}$$

Define T_k as

$$T_k = B_k^TB_k = \begin{pmatrix} \alpha_1^2+\beta_2^2 & \alpha_2\beta_2 & & \\ \alpha_2\beta_2 & \alpha_2^2+\beta_3^2 & \ddots & \\ & \ddots & \ddots & \alpha_k\beta_k \\ & & \alpha_k\beta_k & \alpha_k^2+\beta_{k+1}^2 \end{pmatrix},$$

which is a symmetric tridiagonal matrix. Combining (4.1) and (4.2) we find that T_k is the Ritz–Galerkin projection of A^TA onto subspace span$\{W_k\}$ that is generated by the Lanczos tridiagonalization process of A^TA with starting vector $G^{-1}A^Tb$ under the G-inner product [3, section 5.5].

By the convergence theory of Lanczos tridiagonalization for generalized eigenvalue problem of matrix pair $\{A^TA, G\}$, the eigenvalues of T_k will converge to the generalized eigenvalues of $A^TAz = \xi Gz$ as k increases; see [30] and [19, section 10.1.5]. Note that the ith eigenvalue of T_k is $(\theta_i^{(k)})^2$. The proof is completed. ◻

The convergence behavior of $\theta_i^{(k)}$ is governed by the same Kaniel–Paige–Saad theory as in the standard case for a single matrix; see e.g.[19, section 10.1.5]. It states that we get a faster convergence for those generalized eigenvalues that are at the two ends of the spectrum and well separated. For simplicity of expression, we use ξ_i to denote the ith largest generalized eigenvalue of $A^TAz = \xi Gz$. By Proposition 2.1, it means $\xi_1 = \cdots = \xi_r = 1$, $\xi_i = \gamma_i^2/(\gamma_i^2 + \alpha)$ for $r+1 \leq i \leq r+q$ and $\xi_{r+q+1} = \cdots = \xi_n = 0$. Note that the ξ_i are decreasing and clustered at zero. Thus we can get a faster convergence of $(\theta_i^{(k)})^2$ to some largest generalized eigenvalues ξ_i, while the corresponding generalized eigenvectors z_i are also preferentially approximated by Ritz vectors extracted from span$\{W_k\}$. Since the regularized solution lies in span$\{W_k\}$, this ensures that the dominant information encoded by some leading z_i can be captured by pGKB_SPR while the noisy components are filtered out. To be more precise, we have the following result.

THEOREM 4.1. *Suppose A has full column rank, which means $r+q=n$ in the GSVD of $\{A,L\}$. Let $\sigma_i=1$ for $1\leq i\leq r$. Then the kth regularized solution obtained by pGKB_SPR can be written as*

$$x_k=\sum_{i=1}^{n}f_i^{(k)}\frac{u_{A,i}^Tb}{\sigma_i}z_i \tag{4.3}$$

with filter factors

$$f_i^{(k)}=1-\prod_{j=1}^{k}\frac{(\theta_j^{(k)})^2-\xi_i}{(\theta_j^{(k)})^2},\quad i=1,\ldots,n. \tag{4.4}$$

Proof. Note from (2.4) that $\text{rank}(A)=r+q$, which implies $r+q=n$ if A has full column rank. In this case $\sigma_i>0$ for all $1\leq i\leq n$. Following the second proof of Lemma 3.2, x_k is the solution of

$$\min_{x\in\text{span}\{W_k\}}\|Ax-b\|_2=\min_{Rx\in R\text{span}\{W_k\}}\|AR^{-1}Rx-b\|_2=\min_{\substack{\bar{x}\in\text{span}\{V_k\}\\ x=R^{-1}\bar{x}}}\|\bar{A}\bar{x}-b\|_2,$$

which means that $x_k=R^{-1}\bar{x}_k$ with $\bar{x}_k=\arg\min_{\bar{x}\in\text{span}\{V_k\}}\|\bar{A}\bar{x}-b\|_2$. Since $\{v_i\}_{i=1}^k$ are the right Lanczos vectors generated by the standard GKB process of $\bar{A}$ with starting b, we know $\bar{x}_k$ is the kth CGLS solution applied to the normal equation $\bar{A}^T\bar{A}=\bar{A}^Tb$, where the initial solution is set to $\mathbf{0}$ [25, section 6.3]. Since $\bar{A}=AR^{-1}$ has full column rank, using [53, Property 2.8], the expression of $\bar{x}_k$ is given by

$$\bar{x}_k=(I-\mathcal{R}_k(\bar{A}^T\bar{A}))(\bar{A}^T\bar{A})^{-1}\bar{A}^Tb, \tag{4.5}$$

where $\mathcal{R}_k$ is the so-called Ritz polynomial

$$\mathcal{R}_k(\theta)=\prod_{j=1}^{k}\frac{(\theta_j^{(k)})^2-\theta}{(\theta_j^{(k)})^2}.$$

From the proof of Proposition 2.1 we have $Z^TGZ=D_\alpha$. Let $\bar{Z}=ZD_\alpha^{-1/2}$. Then $\bar{Z}$ is a G-orthonormal matrix, and $\bar{A}=U_AD_AZ^{-1}R^{-1}=U_A(D_AD_\alpha^{-1/2})(R\bar{Z})^{-1}$. Therefore, we have $\bar{A}^T\bar{A}=(R\bar{Z})^{-T}(D_A^TD_AD_\alpha^{-1})(R\bar{Z})^{-1}$. Using the fact that

$$(R\bar{Z})^{-1}(R\bar{Z})^{-T}=[(R\bar{Z})^T(R\bar{Z})]^{-1}=(\bar{Z}^TG\bar{Z})^{-1}=I,$$

we obtain

$$I-\mathcal{R}_k(\bar{A}^T\bar{A})=(R\bar{Z})^{-T}(I-\mathcal{R}_k(D_A^TD_AD_\alpha^{-1}))(R\bar{Z})^{-1}=(R\bar{Z})^{-T}\Lambda(R\bar{Z})^{-1},$$

where $\Lambda=\text{diag}(\{1-\mathcal{R}_k(\xi_i)\}_{i=1}^n)$. Substituting the expression of $I-\mathcal{R}_k(\bar{A}^T\bar{A})$ into (4.5), we get

$$\begin{aligned}\bar{x}_k&=(R\bar{Z})^{-T}\Lambda D_\alpha^{1/2}(D_A^TD_A)^{-1}D_A^TU_A^Tb\\&=R\bar{Z}\Lambda D_\alpha^{1/2}(D_A^TD_A)^{-1}D_A^TU_A^Tb\\&=RZD_\alpha^{-1/2}\Lambda D_\alpha^{1/2}(D_A^TD_A)^{-1}D_A^TU_A^Tb\\&=RZ\Lambda(D_A^TD_A)^{-1}D_A^TU_A^Tb.\end{aligned}$$

Therefore, we finally obtain

$$x_k=R^{-1}\bar{x}_k=\sum_{i=1}^{n}(1-\mathcal{R}_k(\xi_i))\frac{u_{A,i}^Tb}{\sigma_i}z_i,$$

and the filter factors are $f_i^{(k)}=1-\mathcal{R}_k(\xi_i)$ for $i=1,\ldots,n$. ∎

This theorem shows that x_k has a filtered GSVD expansion form. If the k Ritz values $\{(\theta_j^{(k)})^2\}_{j=1}^k$ approximate the first k generalized singular values $\{\xi_i\}_{i=1}^k$ in natural order, i.e., $(\theta_j^{(k)})^2 \approx \xi_i$ for $i = 1,\ldots,k$, from (4.4) we can justify that $f_i^{(k)} \approx 1$ for $i = 1,\ldots,k$ and $f_i^{(k)}$ decreases monotonically to zero for $i = k+1,\ldots,n$. This means that x_k mainly contains the first k dominant GSVD components and filters out the others. Therefore, the pGKB_SPR algorithm exhibits the typical semiconvergence behavior, which means that $\|x_k - x_{\text{true}}\|_2$ first decreases, then as k becomes fairly large, x_k will diverge from x_{true} since too many noisy components are included. The transition point is called the semiconvergence point, at which we get the best regularized solution.

Finally, we give a connection of the pGKB based regularization method with another common regularization method, which is based on the joint bidiagonalization of $\{A, L\}$.

THEOREM 4.2. *If $\alpha = 1$, the solution subspaces generated by the pGKB process of $\{A, M\}$ are the same as those generated by the joint bidiagonalization of $\{A, L\}$.*

Proof. Note that $\sigma_i^2 + \rho_i^2 = 1$ for $r+1 \le i \le r+q$. From (2.10) we have $D_\alpha = I$ when $\alpha = 1$. By Proposition 3.1, we have

$$\mathcal{S}_k = \text{span}\{Z(D_A^T D_A)^i D_A^T U_A^T b\}_{i=0}^{k-1}.$$

By checking the solution subspace given in [36], we find that $\mathcal{S}_k$ is just the subspace generated by the joint bidiagonalization of $\{A, L\}$. □

The joint bidiagonalization based regularization method is effective when L is available. If L is not known or the square root decomposition of M is costly, the pGKB based regularization method is a very good alternative.

5. Hybrid regularization method based on pGKB.

Although the DP or L-curve criterion can be used to stop iteration early to avoid semiconvergence, the corresponding solution is still often over- or under-regularized since the relative error is very sensitive near the semiconvergence point. The hybrid regularization method is another type of iterative method that can stabilize the convergence behavior, which usually applies Tikhonov regularization to the projected problem at each iteration; see, e.g., [12, 37, 47].

By writting $x = W_k y$ with $y \in \mathbb{R}^k$ for any $x \in \mathcal{S}_k$, the Tikhonov regularization problem (1.4) projected onto $\mathcal{S}_k = \text{span}\{W_k\}$ becomes

$$\min_{x=W_k y}\{\|Ax-b\|_2^2 + \lambda x^T M x\} = \min_{y\in\mathbb{R}^k}\{\|B_k y - \beta_1 e_1\|_2^2 + \lambda y^T(W_k^T M W_k)y\}.$$

Note that $W_k^T M W_k \in \mathbb{R}^{k\times k}$. First we compute the square root decomposition $W_k^T M W_k = C_k^T C_k$, which can be done directly by the eigenvalue decomposition of $W_k^T W Z_k$ for small k. The hybrid method solves the regularized projected problem

$$y_k^{\mu_k} = \arg\min_{y\in\mathbb{R}^k}\{\|B_k y - \beta_1 e_1\|^2 + \mu_k \|C_k y\|^2\}, \tag{5.1}$$

and let $x_k^{\mu_k} = W_k y_k^{\mu_k}$ for $k = 1, 2, \ldots$, where μ_k should be determined at each iteration. Let λ^{opt} and μ_k^{opt} denote the optimal regularization parameters of the original problem and kth projected problem, respectively. The main idea of the hybrid method is that as k gradually increases, the projected problem approximates the original problem. Then for a large enough k, we can hope that μ_k^{opt} converges to λ^{opt} and thus the corresponding regularized solution will also converge [37, 47].

To determine μ_k^{opt} at each iteration, we adapt the WGCV, which was first proposed in [12] for standard-form Tikhonov regularization. From (5.1) we have $y_k^{\mu} = (B_k^T B_k + \mu C_k^T C_k)^{-1} B_k^T \beta_1 e_1 := B_{k,\mu}^{\dagger}\beta_1 e_1$. At the kth step, the WGCV method finds the minimizer of the following function with the weight parameter ω_k:

$$G(\omega_k, \mu) = \frac{\|B_k y_k^{\mu} - \beta_1 e_1\|_2^2}{\left(\mathrm{trace}(I - \omega_k B_k B_{k,\mu}^{\dagger})\right)^2}, \tag{5.2}$$

and use this minimizer as μ_k. We remark that if $w_k = 1$ for all k, it is the standard GCV method. The weight parameter ω_k is initialized and automatically updated following the same strategy in [12]. By writing the analytical expression of $G(\omega_k, \mu)$ using the GSVD of $\{B_k, C_k\}$, we can find its minimizer using the MATLAB built-in function `fminbnd.m`. This approach is at a cost of $O(k^3)$ flops, dominated by computing the GSVD of $\{B_k, C_k\}$. In the ideal situation, as k increases, μ_k will converge and $G(1, \mu_k)$ will converge to a fixed value. We terminate the iteration at $k + s_1$ with k the first iteration satisfying

$$\left|\frac{G(1, \mu_{i+1}) - G(1, \mu_i)}{G(1, \mu_1)}\right| < \mathtt{tol1}, \quad i = k, \ldots, k + s_1, \tag{5.3}$$

where $s_1 + 1$ is the length of the window to avoid bumps. We set $s_1 = 4$ and $\mathtt{tol1} = 10^{-6}$ by default.

If we have a good estimate of $\|e\|_2$, there is another method that can update μ_k step by step quickly based on DP. This method is called secant update (SU), which was first proposed for the Arnoldi–Tikhonov hybrid method [17]. Here we show how this method is adapted to update μ_k for the pGKB based hybrid method. A heuristic derivation is as follows. At each iteration, we define the function

$$\psi_k(\mu) = \|A x_k^{\mu} - b\|_2 = \|B_k y_k^{\mu} - \beta_1 e_1\|_2. \tag{5.4}$$

In order to estimate μ_k by DP, we consider determining the proper k and μ_k simultaneously by solving the nonlinear equation $\psi_k(\mu) = \tau\|e\|_2$ with a fixed $\tau > 1$ slightly. We remark that this equation has a solution only when k is sufficiently large. We use the following secant method to solve the above equation. Starting from an initial value μ_0, suppose we already have μ_{k-1} at the $(k-1)$th iteration. Notice that $\psi_k(\mu)$ monotonically increases with respect to μ [40]. It can be approximated by the linear function

$$f(\mu) = \psi_k(0) + \frac{\psi_k(\mu_{k-1}) - \psi_k(0)}{\mu_{k-1}}\mu,$$

which interpolates $\psi_k(\mu)$ at 0 and μ_{k-1}. To update μ_k for the next step, we replace $\psi_k(\mu)$ by $f(\mu)$ and solve $f(\mu) = \tau\|e\|_2$, which leads to

$$\mu_k = \frac{\tau\|e\|_2 - \psi_k(0)}{\psi_k(\mu_{k-1}) - \psi_k(0)}\mu_{k-1}.$$

This update formula may suffer from instability for small k, since it holds that $\psi_k(0) > \tau\|e\|_2$. Therefore, we finally use

$$\mu_k = \left|\frac{\tau\|e\|_2 - \psi_k(0)}{\psi_k(\mu_{k-1}) - \psi_k(0)}\right|\mu_{k-1} \tag{5.5}$$

Algorithm 5.1. pGKB_HR.

```
Input: $A$, $b$, $M$, $\alpha > 0$, $\omega_1$ or $\mu_0$          ▷ require $\tau\|e\|_2$ for SU
for $k = 1, 2, \ldots,$ do
    Compute $u_k$, $w_k$, $\alpha_k$, $\beta_k$ by pGKB
    Compute decomposition $W_k^T M W_k = C_k^T C_k$ to get $C_k$
    if 'hybrid = WGCV' then                                   ▷ WGCV
        Compute the GSVD of $\{B_k, C_k\}$
        Determine $\mu_k$ by minimizing (5.2)
        Compute $y_k^{\mu_k}$ by solving (5.1)
        Update $\omega_k$ following [12]
    else if 'hybrid = SU' then                                ▷ SU
        Compute $y_k^{\mu_{k-1}}$ by solving (5.1)
        Compute the residual norm $\psi_k(\mu_{k-1})$ by (5.4)
        Computing $\bar{\phi}_{k+1}$ recursively by Algorithm 3.2, lines 3–6
        Update $\mu_k$ by (5.5)
    end if
    Terminate iteration by (5.3) or (5.6) at $k_2$
end for
Compute $x_{k_2}^{\mu_{k_2}} = W_{k_2} y_{k_2}^{\mu_{k_2}}$
Output: Final regularized solution $x_{k_2}$
```

as the practical formula to update μ_k. We set $\mu_0 = 1.0$ by default. Numerically we find that (5.5) is very stable in the sense that when k is sufficiently large, both the regularization parameter μ_k and residual norm $\psi_k(\mu_{k-1})$ tend to plateau.

Since $\psi_k(0) = \|Ax_k - b\|_2 = \bar{\phi}_{k+1}$, which is the residual norm of the kth pGKB_SPR solution, it can be updated efficiently by Algorithm 3.2 (without computing p_k and x_k). To terminate the iteration, we choose $k + s_2$ as the stopping iteration with k the first iteration satisfying

$$\psi_k(0) \leq \tau\|e\|_2 \quad \text{and} \quad \left|\frac{\psi_{i+1}(\mu_i) - \psi_i(\mu_{i-1})}{\psi_i(\mu_{i-1})}\right| \leq \texttt{tol2}, \quad i = k, \ldots, k + s_2, \tag{5.6}$$

where $s_2 + 1$ is the length of the window to avoid bumps. We set $s_2 = 4$ and $\texttt{tol2} = 0.001$ by default.

To summarize, we show the pseudocode of the pGKB based hybrid regularization (pGKB_HR) algorithm using WGCV or SU in Algorithm 5.1.

6. Experimental results.

We use some numerical examples to show the effectiveness and performance of the proposed algorithms, including pGKB_SPR with DP and the L-curve as early stopping criteria and pGKB_HR with WGCV and SU for determining μ_k. All experiments are performed with MATLAB R2019b. The codes are available at https://github.com/Machealb/InverProb_IterSolver, where some codes in the packages of [16, 26] are exploited. We compare the accuracy of the regularized solutions and show convergence behaviors by using the relative reconstruction error

$$RE(k) = \frac{\|x_k - x_{\text{true}}\|}{\|x_{\text{true}}\|} \quad \text{or} \quad RE(k) = \frac{\|x_k^{\mu_k} - x_{\text{true}}\|}{\|x_{\text{true}}\|} \tag{6.1}$$

to plot the convergence curve of each algorithm.

6.1. Small-scale inverse problems.

For the first test problem, we choose deriv2 from [26] by setting $m = n = 2000$, which is a discretization of the first kind Fredholm integral equation

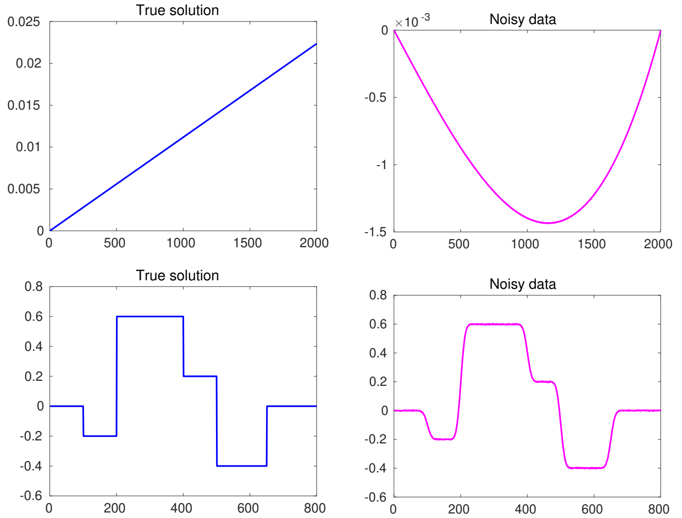


FIG. 1. *Illustration of the true solution and noisy observed data. Top: true and observed functions for* deriv2*. Bottom: true and noisy deblurred signals for* gauss1d*. Note: color appears only in the online article.*

$$b(s)=\int_0^1 K(s,t)x(t)\mathrm{d}t, \quad K(s,t)=\begin{cases} s(t-1), & s<t, \\ t(s-1), & s\geq t, \end{cases} \tag{6.2}$$

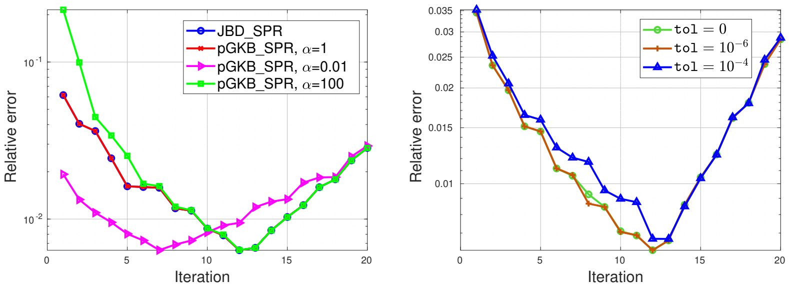
where $x(t)=t$ and $(s,t)\in[0,1]^2$. For the second test problem, we use the Gaussian convolution of a one-dimensional (1D) piecewise constant signal

$$b(s)=\int_{-\infty}^{+\infty} K(s-t)x(t)\mathrm{d}t, \quad K(s-t)=\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(s-t)^2}{2\sigma^2}\right), \tag{6.3}$$

where K is the Gaussian kernel with $\sigma=10$, and the discretized signal $x(t)$ on 800 uniform grids over $[0,1]$ is shown in Figure 1. The Gaussian kernel K is discretized correspondingly with zero boundary condition such that $A\in\mathbb{R}^{800\times 800}$ is a Toeplitz matrix. This problem is named as gauss1d. For each problem, we add a Gaussian white noise with noise level $\varepsilon=\|e\|_2/\|b_{\mathrm{true}}\|_2$ to b_{true} and form $b=b_{\mathrm{true}}+e$, where we set $\varepsilon=5\times 10^{-4}$ and $\varepsilon=5\times 10^{-3}$ for deriv2 and gauss1d, respectively. The true solutions and noisy observed data for these two problems are shown in Figure 1.

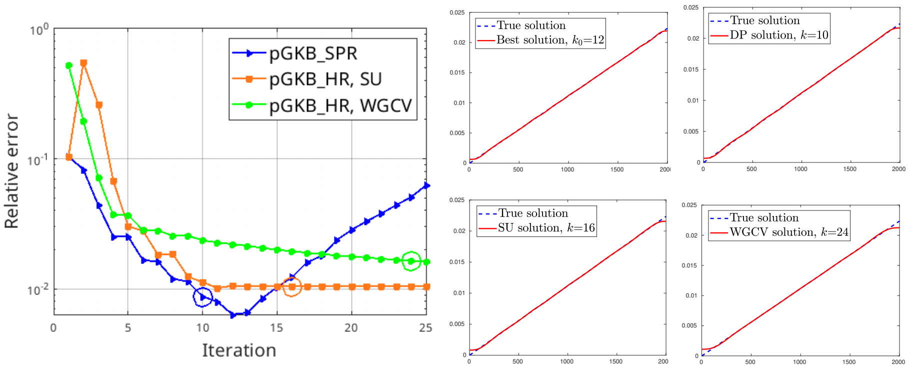
For deriv2, we set the regularization matrix $L\in\mathbb{R}^{(n-1)\times n}$ as the scaled discretization of the 1D differential operator, which is a bidiagonal matrix with one more row than columns and values -1 and 1 on the subdiagonal and diagonal parts, respectively. Then we let $M=L^TL$ to get M. Since L is available, we can compare pGKB_SPR with the joint bidiagonalization (JBD) based subspace projection regularization (JBD_SPR) algorithm.

First, we illustrate the regularization effect of pGKB_SPR by plotting its semiconvergence curve and comparing it with JBD_SPR. In this experiment, the inner iteration is computed accurately using matrix inversion, and the values of α are set as 1, 0.001 and 100. From Figure 2(a), we find that pGKB_SPR exhibits typical semiconvergence behavior, and the relative errors at the semiconvergence point for the three different α's are the same as that of JBD_SPR. This confirms that the pGKB based regularization method has a good regularization effect. To illustrate the



(a) $\tau = 0$ (b) $\alpha = 10$

FIG. 2. *Comparison of semiconvergence curves between JBD and pGKB based subspace projection regularization algorithms for* deriv2. (a) *Comparison for different* α*'s where inner iterations are computed accurately.* (b) *Comparison for different solution accuracies of inner iterations. Note: color appears only in the online article.*



(a) Convergence history (b) Reconstructed solution

FIG. 3. *The relative error curves of regularized solutions computed by the pGKB based algorithms and corresponding reconstructed solutions for* deriv2. *Note: color appears only in the online article.*

impact of computing accuracy of inner iteration on the accuracy of the regularized solution, for pGKB_SPR with $\alpha = 10$, we use the MATLAB function `pcg.m` to solve $Gs = A^T u_i$ at each outer iteration with stopping tolerance $\texttt{tol} = 10^{-6}, 10^{-4}$, and plot the semiconvergence curves in Figure 2(b). We remark that $\texttt{tol} = 0$ means that the inner iterations are computed accurately using matrix inversion. We find that for $\texttt{tol} = 10^{-6}$ the best regularized solution has the same relative error as that for $\texttt{tol} = 0$, and the two curves almost coincide for many steps even after semiconvergence. For a much lower computing accuracy with $\texttt{tol} = 10^{-4}$, there is a loss of accuracy for the regularized solution. The required solution accuracy of the inner iteration affects the accuracy of the algorithm, but a theoretical analysis of it is complicated. This will be a part of our forthcoming work.

Then we illustrate the convergence behavior of pGKB_SPR and pGKB_HR, where DP, L-curve (LC) are used to stop iteration early and WGCV, SU are used to determine μ_k. In this experiment, we set $\alpha = 10$ and $\texttt{tol} = 10^{-6}$. The convergence history curves and the corresponding reconstruct solutions are shown in Figure 3. The stopping iterations of pGKB_SPR with DP and pGKB_HR with SU and WGCV

TABLE 1

Relative errors of the final regularized solutions and corresponding early stopping iterations (in parentheses), where $\varepsilon = 5 \times 10^{-4}$ for deriv2 *and $\varepsilon = 5 \times 10^{-3}$ for* gauss1d.

Problem	Best	DP	LC	SU	WGCV
deriv2	0.0064 (12)	0.0087 (10)	0.0120 (8)	0.0105 (16)	0.0165 (24)
gauss1d	2.2395×10^{-4} (4)	3.0393×10^{-4} (3)	5.6806×10^{-4} (5)	6.4605×10^{-4} (11)	6.1523×10^{-4} (25)

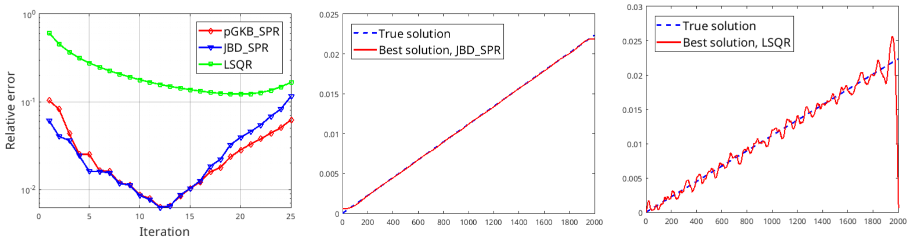


FIG. 4. *Comparison of regularization effect between LSQR, JBD, and pGKB based algorithms for* deriv2*: relative error curves and the best reconstructed solutions by JBD_SPR and LSQR. Note: color appears only in the online article.*

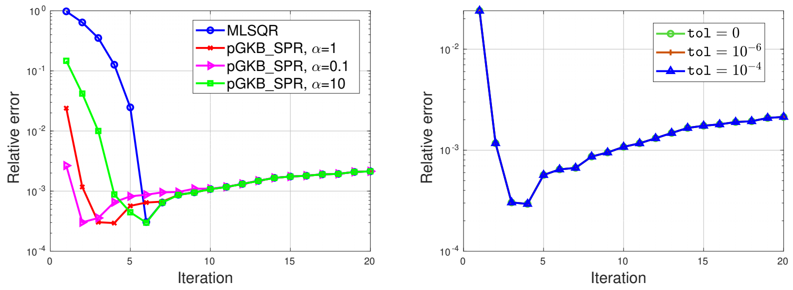
are marked by circles in the relative error curves. Note that the reconstructed solution for pGKB_SPR with LC is omitted since it is similar to that for pGKB_SPR with DP. The relative errors of the final regularized solutions and the corresponding iteration number are shown in Table 1. Although both DP and LC underestimate the semiconvergence point k_0, the estimated early stopping iterations do not deviate far from k_0, and the reconstructed solutions approximate well to the true solution. For pGKB_HR with WGCV and SU, the relative error eventually decays flat as k becomes sufficiently large, and is slightly higher than the best relative error of pGKB_SPR at k_0. Although it is not shown here, in this experiment we find that both WGCV and SU overestimate regularization parameters, and the estimate by WGCV is higher, thereby it computes a more oversmoothed solution.

We use Figure 4 to compare the regularization effect between LSQR, JBD, and pGKB based algorithms for deriv2, where we set $\alpha = 10$, and use `tol` $= 10^{-6}$ for inner iterations of both JBD and pGKB. We can clearly find that LSQR is a worse choice for reconstructing a good solution. In contrast, the JBD and pGKB based algorithms compute solutions with similar high accuracies, since both of them efficiently incorporate prior information about x_{true} encoded by L or M.

For gauss1d with a piecewise constant x_{true}, the total variation regularization $\text{TV}(x) = \int_{\mathbb{R}} |\nabla x| \text{d}t$ is the most suitable. Since $\text{TV}(x)$ is nonlinear, we construct M to approximate $\text{TV}(x)$ at x_{true} by $x^T M x$ using the procedure in the LDFP method. We remark that this M is only used for experimental purposes, while in practice we can not construct such a good M since x_{true} is unknown. The LDFP method first replaces $\text{TV}(x)$ by $\text{TV}_\beta(x) = \int_{\mathbb{R}} \sqrt{|\nabla x|^2 + \beta^2} \text{d}t$ with β a small positive value, and then linearizes TV_β at x using its gradient $L(x)$:

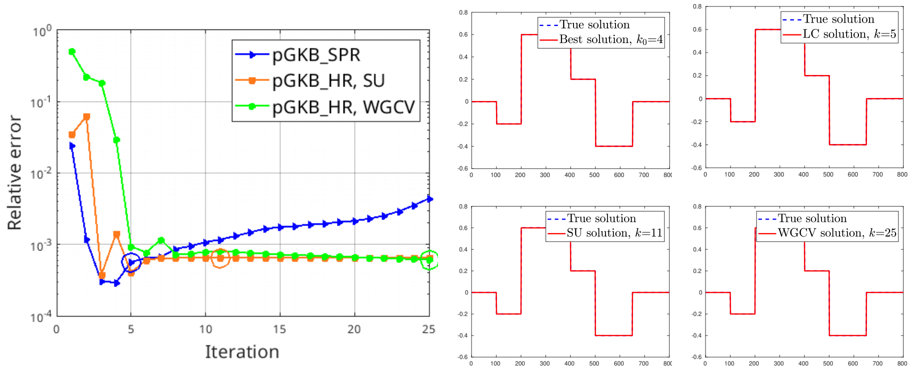
$$L(x)y := -\nabla \cdot \left(\frac{1}{\sqrt{|\nabla x|^2 + \beta^2}} \nabla y \right). \tag{6.4}$$

We choose $\beta = 10^{-6}$ and construct M by discretizing $L(x)$ at x_{true} using the finite difference procedure. For details, see [11, 56].



(a) $\tau = 0$ (b) $\alpha = 1$

FIG. 5. *Comparison of semiconvergence curves between MLSQR and pGKB based subspace projection regularization algorithms for* gauss1d. (a) *Comparison for different* α*'s where inner iterations are computed accurately.* (b) *Comparison for different solution accuracies of inner iterations. Note: color appears only in the online article.*



(a) Convergence history (b) Reconstructed solution

FIG. 6. *The relative error curves of regularized solutions computed by the pGKB based algorithms and corresponding reconstructed solutions for* gauss1d. *Note: color appears only in the online article.*

In this experiment, we first compare the pGKB method with the MLSQR method proposed in [1]. Since the rank of M is $n-1$, we let $M_\delta = M + \delta I$ with $\delta = 10^{-6}$ and apply MLSQR to $\{A, M_\delta\}$. The semiconvergence curves are shown in Figure 5(a), where the inner iteration is computed accurately using matrix inversion, and the values of α are set as 1, 0.01, and 10. We find that pGKB_SPR exhibits typical semiconvergence behavior, and the relative errors at the semiconvergence point for the three different α's are almost the same as that of MLSQR. The impact of computing the accuracy of an inner iteration on the accuracy of the regularized solution for pGKB_SPR is shown in Figure 5(b), where we use $\alpha = 1$. All three curves almost coincide for many steps even after semiconvergence. This indicates that a much relaxed computing accuracy such as $\texttt{tol} = 10^{-4}$ for inner iterations does not reduce the accuracy of the final regularized solution.

The convergence behavior of pGKB_SPR and pGKB_HR algorithms and the corresponding reconstructed 1D signals are shown in Figure 6, where $\alpha = 1$ and $\texttt{tol} = 10^{-6}$. The stopping iterations of pGKB_SPR with LC and pGKB_HR with SU and WGCV are marked by circles in the relative error curves. The relative errors of the final regularized solutions and the corresponding iteration numbers are shown in

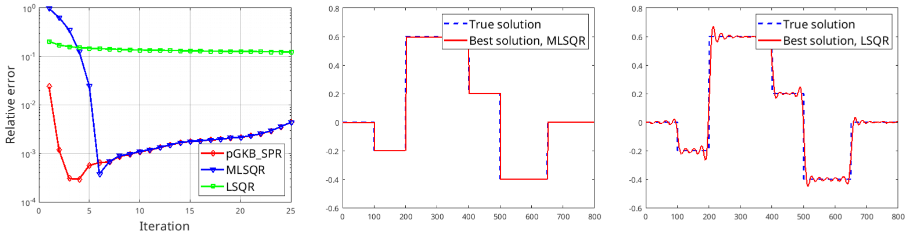


FIG. 7. *Comparison of regularization effect between LSQR, MLSQR, and pGKB based algorithms for* gauss1d*: relative error curves and the best reconstructed solutions by MLSQR and LSQR. Note: color appears only in the online article.*

Table 1. For pGKB_HR with WGCV and SU, the relative errors eventually decay flat as k becomes sufficiently large and they are very close. Note that the iteration of WGCV does not satisfy the stopping criterion (5.3) when the maximum iteration number 25 is reached, since the default value for `tol1` is too small in this case. All the reconstructed signals approximate well to the original piecewise constant signal. This is due to the well-constructed regularizer matrix M based on TV regularization and the incorporation of prior information about x_{true} in the solution subspace span$\{W_k\}$. Further comparison of the regularization effect between LSQR, MLSQR, and pGKB based algorithms for gauss1d is shown in Figure 7, where we set $\alpha = 1$ for pGKB, $\delta = 10^{-6}$ for MLSQR, and use $\texttt{tol} = 10^{-6}$ for inner iterations of both MLSQR and pGKB. Clearly, LSQR is a worse choice for reconstructing a good solution, while, in contrast, both MLSQR and pGKB based algorithms exhibit very good regularization effects.

6.2. Large-scale inverse problems. The two large-scale test problems are chosen from [16]. The first test problem is PRblurdefocus PRblurdefocus, which models an image blurring problem caused by a spatially invariant out-of-focus blur. We use the true image "Hubble Space Telescope" with 256×256 pixels, and set the blur level as "`mild`" and use the zero boundary condition to get $A \in \mathbb{R}^{256^2 \times 256^2}$. The second test problem is PRdiffusion, which is a two-dimensional (2D) inverse diffusion problem $\partial_t u = \nabla^2 u$ in the domain $[0, T] \times [0, 1]^2$ with Neumann boundary condition, and we aim to reconstruct the initial function u_0 from u_T. We set $T = 0.005$ with 100 time steps, and discretize the PDE on a 128×128 uniform finite element mesh to get the forward operator $A \in \mathbb{R}^{128^2 \times 128^2}$, which maps x_{true} (the discretized u_0) to b_{true} (the discretized u_T). The noise levels are set as $\varepsilon = 0.002$ and $\varepsilon = 0.001$ for PRblurdefocus and PRdiffusion, respectively. The true solutions and noisy observed data are shown in Figure 8.

For PRblurdefocus, we use the same procedure as for gauss1d to construct M. Here the deblurring matrix A is an object so that only the matrix-vector products Av or $A^T v$ are available. Therefore, we can only compute the inner iteration by iteratively solving $Gs = A^T u_i$. We set $\alpha = 0.1$ and use `pcg.m` with stopping tolerance $\texttt{tol} = 10^{-6}$ to compute inner iterations. We compare the convergence behaviors of pGKB_SPR and pGKB_HR by plotting their convergence history curves. The convergence history curves and the corresponding reconstructed solutions are shown in Figure 9, where the stopping iterations of pGKB_SPR with DP and pGKB_HR with SU and WGCV are marked by circles in the relative error curves. The relative errors of the final regularized solutions and the corresponding iteration numbers

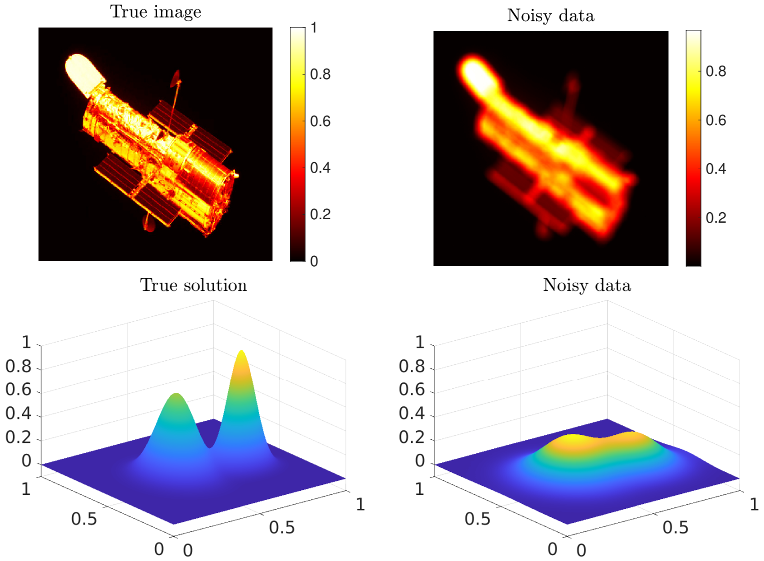
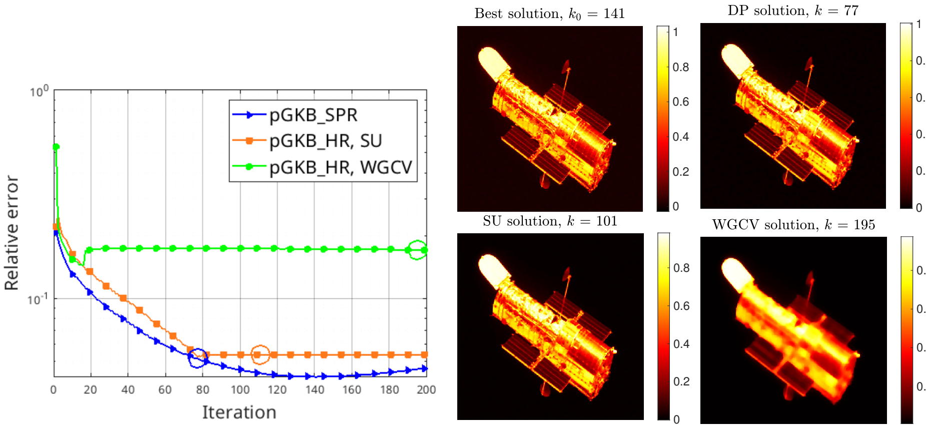


FIG. 8. *Illustration of the true solution and noisy observed data. Top: true and noisy blurred image for* PRblurdefocus*. Bottom: x_{true} corresponding to u_0 and b corresponding to noisy u_T for* PRdiffusion*. Note: color appears only in the online article.*



(a) Convergence history (b) Reconstructed solution

FIG. 9. *The relative error curves of regularized solutions computed by the pGKB based algorithms and corresponding reconstructed images for* PRblurdefocus*. Note: color appears only in the online article.*

are shown in Table 2. From Figure 9(a) we observe the typical semiconvergence behavior of pGKB_SPR, and both DP and LC underestimate the semiconvergence point k_0. For pGKB_HR with SU, the relative error gradually decreases to a constant value for sufficiently large k, and it is slightly higher than the relative error of pGKB_SPR at k_0. For pGKB_HR with WGCV, the relative error eventually stagnates at a value much higher than that of pGKB_SPR at k_0, since WGCV significantly overestimates the regularization parameter. The corresponding reconstructed images are shown in Figure 9(b), where we can clearly see that the reconstruction quality

Table 2
Relative errors of the final regularized solutions and corresponding early stopping iterations (in parentheses), where $\varepsilon = 0.002$ for PRblurdefocus *and $\varepsilon = 0.001$ for* PRdiffusion.

Problem	Best	DP	LC	SU	WGCV
PRblurdefocus	0.0422 (141)	0.0515 (77)	0.0508 (79)	0.0539 (101)	0.1717 (195)
PRdiffusion	0.0497 (90)	0.0602 (71)	0.0526 (83)	0.0578 (93)	0.0946 (91)

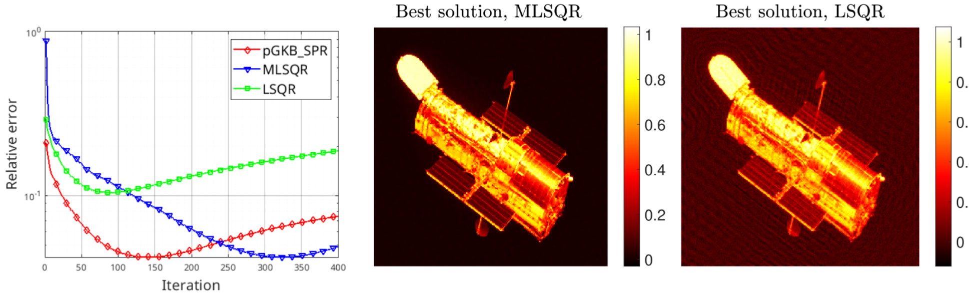
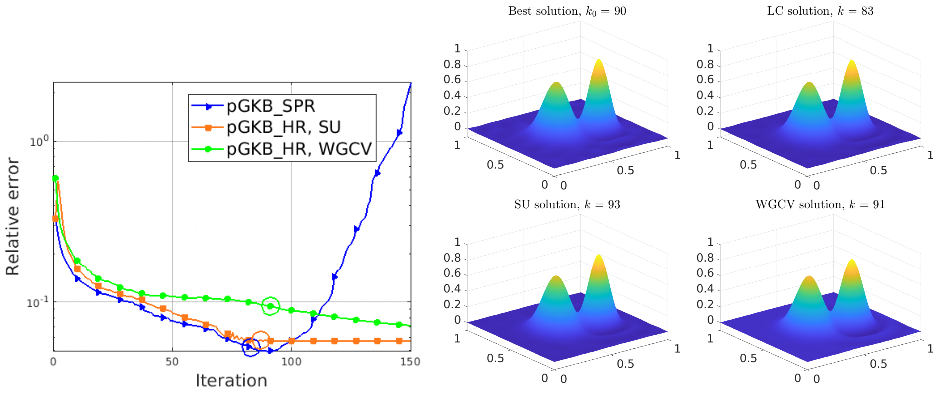


Fig. 10. *Comparison of regularization effect between LSQR, MLSQR, and pGKB based algorithms for* PRblurdefocus*: relative error curves and the best reconstructed images by MLSQR and LSQR. Note: color appears only in the online article.*

for WGCV is much poorer than other algorithms. This is because the estimated μ_k by WGCV does not converge to a good regularization parameter of the original problem (1.4).

To further illustrate the regularization effect of the pGKB based algorithm, we depict together the relative error curves of pGKB_SPR, MLSQR, and LSQR in Figure 10, where we set $\delta = 10^{-8}$ for MLSQR and also use `tol` $= 10^{-6}$ for inner iterations of MLSQR. It is clear that MLSQR and pGKB_SPR achieve similar relative errors at the semiconvergence point, much smaller than that obtained by LSQR. Correspondingly, the best reconstructed image by LSQR is much worse than those by MLSQR and pGKB_SPR. This is attributed to the effective treatment of the regularization matrix M by MLSQR or pGKB_SPR.

For PRdiffusion, we set M as the discretized 2D negative Laplacian to enforce some smoothness on the desired initial u_0. Here the forward operator matrix A is a function handle that represents the computation process of the numerical solution of $\partial_t u = \nabla^2 u$. Therefore, we can only compute the inner iteration by solving $Gs = A^T u_i$ using an iterative solver that is based on matrix-vector products. We set $\alpha = 1$ in pGKB and use `pcg.m` with stopping tolerance `tol` $= 10^{-6}$ to compute inner iterations. The relative errors, the marked stopping iterations for pGKB_SPR with LC and pGKB_HR with SU and WGCV, and the reconstructed solutions are shown in Figure 11. We can find from Figure 11 and Table 2 that both DP and LC slightly underestimate k_0 for pGKB_SPR, and both the corresponding reconstructed solutions are of high quality. For pGKB_HR, as k becomes sufficiently large, both WGCV and SU compute iterative solutions with relative errors decreasing toward a value slightly larger than the best for pGKB_SPR, and the corresponding final regularized solutions approximate well to x_{true}. The comparison of regularization effects between pGKB_SPR, MLSQR, and LSQR is shown in Figure 12. Since M is invertible, here we do not need to set a δ for MLSQR. Although this M may not be the best regularization matrix for PRdiffusion, we still find that the best regularized solution computed by pGKB_SPR has similar



(a) Convergence history (b) Reconstructed solution

Fig. 11. *The relative error curves of regularized solutions computed by the pGKB based algorithms and corresponding reconstructed solutions for* PRdiffusion*. Note: color appears only in the online article.*

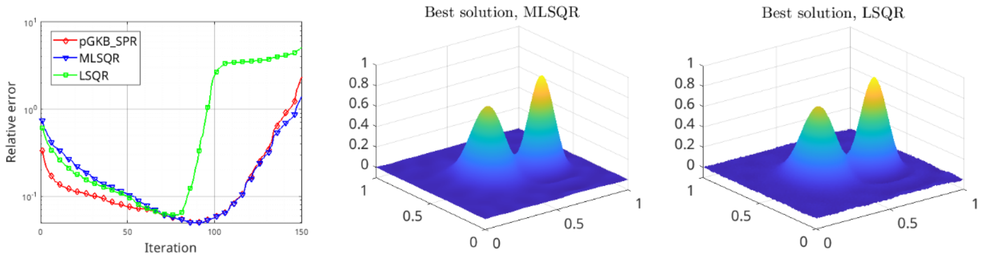


Fig. 12. *Comparison of regularization effect between LSQR, MLSQR, and pGKB based algorithms for* PRdiffusion*: relative error curves and the best reconstructed solutions by MLSQR and LSQR. Note: color appears only in the online article.*

accuracy to that by MLSQR, where the solution by LSQR is slightly less accurate. This confirms the good regularization effect of the pGKB based algorithm.

7. Conclusion and outlook. For linear inverse problems with general-form Tikhonov regularization term x^TMx, where M is positive semidefinite, we have proposed several iterative regularization algorithms. These algorithms are based upon a new iterative process called the pGKB that implicitly utilizes a proper preconditioner to generate solution subspaces incorporating prior properties of the desired solution. The pGKB_SPR algorithm is proposed, where the DP or LC is used as an early stopping criterion. The regularization effect of pGKB_SPR is analyzed by showing that the iterative solution has a filtered GSVD expansion form, thus revealing the semiconvergence behavior of it. To avoid semiconvergence of pGKB_SPR, two pGKB_HR algorithms are proposed that adopt WGCV and SU for determining regularization parameters at each iteration, respectively. Both small-scale and large-scale linear inverse problems are used to test the proposed algorithms and illustrate their excellent effectiveness and performance.

There are several issues remaining to be further studied. For example, numerical results indicate that the accuracy of an inner iteration in pGKB may be moderately relaxed without compromising the accuracy of the final regularized solution, raising the need for theoretical analysis about the required accuracy of inner iteration. Another issue concerns the SU method for pGKB_HR, which numerically exhibits good convergence for iterative solutions. Thus it is necessary to analyze the convergence behaviors of the regularization parameter μ_k and iterative solution.

Acknowledgments. The author is grateful to the anonymous referees and editor Prof. Melina Freitag for their detailed reading of the manuscript and insightful comments that helped to improve the paper. The author thanks Dr. Long Wang and Prof. Weile Jia for their consistent encouragement and support during this research.

REFERENCES

[1] S. R. Arridge, M. M. Betcke, and L. Harhanen, *Iterated preconditioned LSQR method for inverse problems on unstructured grids*, Inverse Problems, 30 (2014), 075009.

[2] X. Bai, G.-X. Huang, X.-J. Lei, L. Reichel, and F. Yin, *A novel modified TRSVD method for large-scale linear discrete ill-posed problems*, Appl. Numer. Math., 164 (2021), pp. 72–88.

[3] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst, *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*, SIAM, Philadelphia, 2000.

[4] G. Bin, S. Wu, M. Shao, Z. Zhou, and G. Bin, *IRN-MLSQR: An improved iterative reweight norm approach to the inverse problem of electrocardiography incorporating factorization-free preconditioned LSQR*, J. Electrocardiol., 62 (2020), pp. 190–199.

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

[6] T. Buzug, *Computed Tomography*, Springer, Berlin, 2008, https://doi.org/10.1007/978-3-540-39408-2.

[7] D. Calvetti, *Preconditioned iterative methods for linear discrete ill-posed problems from a Bayesian inversion perspective*, J. Comput. Appl. Math., 198 (2007), pp. 378–395.

[8] D. Calvetti, F. Pitolli, E. Somersalo, and B. Vantaggi, *Bayes meets Krylov: Statistically inspired preconditioners for CGLS*, SIAM Rev., 60 (2018), pp. 429–461.

[9] D. Calvetti and E. Somersalo, *Priorconditioners for linear systems*, Inverse Problems, 21 (2005), pp, 1397–1418.

[10] V. Candiani, N. Hyvönen, J. P. Kaipio, and V. Kolehmainen, *Approximation error method for imaging the human head by electrical impedance tomography*, Inverse Problems, 37 (2021), 125008.

[11] T. F. Chan and P. Mulet, *On the convergence of the lagged diffusivity fixed point method in total variation image restoration*, SIAM J. Numer. Anal., 36 (1999), pp. 354–367.

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

[13] J. Chung and A. K. Saibaba, *Generalized hybrid iterative methods for large-scale Bayesian inverse problems*, SIAM J. Sci. Comput., 39 (2017), pp. S24–S46.

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

[15] H. W. Engl, M. Hanke, and A. Neubauer, *Regularization of Inverse Problems*, Kluwer Academic, Dordrecht, 2000.

[16] S. Gazzola, P. C. Hansen, and J. G. Nagy, *IR tools: A MATLAB package of iterative regularization methods and large-scale test problems*, Numer. Algorithms, 81 (2019), pp. 773–811.

[17] S. Gazzola and P. Novati, *Automatic parameter setting for Arnoldi–Tikhonov methods*, J. Comput. Appl. Math., 256 (2014), pp. 180–195.

[18] M. G. Genton, *Classes of kernels for machine learning: A statistics perspective*, J. Mach. Learn. Res., 2 (2001), pp. 299–312.

[19] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 4th ed., The Johns Hopkins University Press, Baltimore, MD, 2013.

[20] G. H. Golub and H. G. Wahba, *Generalized cross-validation as a method for choosing a good ridge parameter*, Technometrics, 21 (1979), pp. 215–223.

[21] M. Hanke, J. Nagy, and R. Plemmons, *Preconditioned iterative regularization for ill-posed problems*, in Numerical Linear Algebra, De Gruyter, Berlin, 1993, pp. 141–164.

[22] A. Hannukainen, L. Harhanen, N. Hyvönen, and H. Majander, *Edge-promoting reconstruction of absorption and diffusivity in optical tomography*, Inverse Problems, 32 (2016), 015008.

[23] A. Hannukainen, N. Hyvönen, H. Majander, and T. Tarvainen, *Efficient inclusion of total variation type priors in quantitative photoacoustic tomography*, SIAM J. Imaging Sci., 9 (2016), pp. 1132–1153.

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

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

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

[27] P. C. Hansen, *Discrete Inverse Problems: Insight and Algorithms*, SIAM, Philadelphia, 2010.

[28] P. C. Hansen, J. G. Nagy, and D. P. O'Leary, *Deblurring Images: Matrices, Spectra and Filtering*, SIAM, Philadelphia, 2006.

[29] L. Harhanen, N. Hyvönen, H. Majander, and S. Staboulis, *Edge-enhancing reconstruction algorithm for three-dimensional electrical impedance tomography*, SIAM J. Sci. Comput., 37 (2015), pp. B60–B78.

[30] U. L. Hetmaniuk and R. B. Lehoucq, *Uniform accuracy of eigenpairs from a shift–invert Lanczos method*, SIAM J. Matrix Anal. Appl., 28 (2006), pp. 927–948, https://doi.org/10.1137/050629288.

[31] G. Huang, Y. Liu, and F. Yin, *Tikhonov regularization with MTRSVD method for solving large-scale discrete ill-posed problems*, J. Comput. Appl. Math., 405 (2022), 113969.

[32] G. Huang, L. Reichel, and F. Yin, *Projected nonstationary iterated Tikhonov regularization*, BIT, 56 (2016), pp. 467–487.

[33] Z. Jia and H. Li, *The joint bidiagonalization method for large GSVD computations in finite precision*, SIAM J. Matrix Anal. Appl., 44 (2023), pp. 382–407.

[34] Z. Jia and Y. Yang, *A joint bidiagonalization based iterative algorithm for large scale general-form Tikhonov regularization*, Appl. Numer. Math., 157 (2020), pp. 159–177.

[35] J. Kaipio and E. Somersalo, *Statistical and Computational Inverse Problems*, Springer, New York, 2006.

[36] 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.

[37] M. E. Kilmer and D. P. O'Leary, *Choosing regularization parameters in iterative methods for ill-posed problems*, SIAM J. Matrix Anal. Appl., 22 (2001), pp. 1204–1221.

[38] K. Z. Kody Law and A. Stuart, *Data Assimilation: A Mathematical Introduction*, Springer, Cham, Switzerland, 2015, https://doi.org/10.1007/978-3-319-20325-6.

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

[40] B. Lewis and L. Reichel, *Arnoldi–Tikhonov regularization methods*, J. Comput. Appl. Math., 226 (2009), pp. 92–102.

[41] H. Li, *The joint bidiagonalization of a matrix pair with inaccurate inner iterations*, SIAM J. Matrix Anal. Appl., 45 (2024), pp. 232–259.

[42] V. A. Morozov, *Regularization of incorrectly posed problems and the choice of regularization parameter*, USSR Comput. Math. Math. Phys., 6 (1966), pp. 242–251.

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

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

[45] P. Perona and J. Malik, *Scale-space and edge detection using anisotropic diffusion*, IEEE Trans. Pattern Anal., 12 (1990), pp. 629–639.

[46] L. Reichel, F. Sgallari, and Q. Ye, *Tikhonov regularization based on generalized Krylov subspace methods*, Appl. Numer. Math., 62 (2012), pp. 1215–1228.

[47] R. A. Renaut, S. Vatankhah, and V. E. Ardesta, *Hybrid and iteratively reweighted regularization by unbiased predictive risk and weighted GCV for projected systems*, SIAM J. Sci. Comput., 39 (2017), pp. B221–B243.

[48] M. Richter, *Inverse Problems: Basics, Theory and Applications in Geophysics*, Springer, Cham, Switzerland, 2016, https://doi.org/10.1007/978-3-319-48384-9.

[49] L. Roininen, M. S. Lehtinen, S. Lasanen, M. Orispää, and M. Markkanen, *Correlation priors*, Inverse Probl. Imaging, 5 (2011), pp. 167–184.

[50] L. I. Rudin, S. Osher, and E. Fatemi, *Nonlinear total variation based noise removal algorithms*, Phys. D, 60 (1992), pp. 259–268.

[51] A. M. Stuart, *Inverse problems: A Bayesian perspective*, Acta Numer., 19 (2010), pp. 451–559.

[52] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems*, Winston, Washington, DC, 1977.

[53] A. Van der Sluis and H. A. van der Vorst, *The rate of convergence of conjugate gradients*, Numer. Math., 48 (1986), pp. 543–560.

[54] C. F. Van Loan, *Generalizing the singular value decomposition*, SIAM J. Numer. Anal., 13 (1976), pp. 76–83.

[55] S. Vatankhah, R. A. Renaut, and V. E. Ardestani, *Total variation regularization of the 3-D gravity inverse problem using a randomized generalized singular value decomposition*, Geophys J. Int., 213 (2018), pp. 695–705.

[56] C. R. Vogel and M. E. Oman, *Iterative methods for total variation denoising*, SIAM J. Sci. Comput., 17 (1996), pp. 227–238.

[57] Y. Wei, P. Xie, and L. Zhang, *Tikhonov regularization and randomized GSVD*, SIAM J. Matrix Anal. Appl., 37 (2016), pp. 649–675.

[58] H. Xiang and J. Zou, *Randomized algorithms for large-scale inverse problems with general Tikhonov regularizations*, Inverse Problems, 31 (2015), 085008.