Inexact Generalized Golub-Kahan Methods for Large-Scale Inverse Problems

Yutong Bu[†]

Project advisor: Julianne Chung[‡]

Abstract. Solving large-scale Bayesian inverse problems presents significant challenges, particularly when the exact (discretized) forward operator is unavailable. These challenges often arise in image processing tasks due to unknown defects in the forward process that may result in varying degrees of inexactness in the forward model. Moreover, for many large-scale problems, computing the square root or inverse of the prior covariance matrix is infeasible such as when the covariance kernel is defined on irregular grids or is accessible only through matrix-vector products. This paper introduces an efficient approach by developing an inexact generalized Golub-Kahan decomposition that can incorporate varying degrees of inexactness in the forward model to solve large-scale generalized Tikhonov regularized problems. Further, a hybrid iterative projection scheme is developed to automatically select Tikhonov regularization parameters. Numerical experiments on simulated tomography reconstructions demonstrate the stability and effectiveness of this novel hybrid approach.

Key words. inverse problems, Krylov subspace methods, Tikhonov regularization

MSC codes. 65F22 65F10 65K10 15A29

1. Introduction.

Inverse problems are prevalent in many scientific applications, but they are usually hard to solve due to their large scale. Inverse problems are typically ill-posed, whereby the number of unknown parameters can be significantly larger than the size of the observed datasets. This projects aims to provide an efficient method to solve large-scale inverse problem through a Bayesian approach, specifically when the exact forward model is not available, as described in [27, 10], but where an approximate model can be obtained.

Consider a linear inverse problem of the form

$$\mathbf{A}\mathbf{s} + \boldsymbol{\epsilon} = \mathbf{d}, \tag{1.1}$$

where $\mathbf{d} \in \mathbb{R}^m$ is the observed data, $\mathbf{A} \in \mathbb{R}^{m\times n}$ is an ill-posed matrix representing the discretized forward model, $\mathbf{s} \in \mathbb{R}^n$ is the desired parameters, and $\boldsymbol{\epsilon} \in \mathbb{R}^m$ is the additive Gaussian noise in the data. Assume $\mathbf{A}$ is only accessible through matrix-vector products (MVPs) with some potential inexactness involved, $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$ where $\mathbf{R}$ is a symmetric positive definite (SPD) matrix with inexpensive inverse and square root, for example a diagonal matrix with positive entries. The goal is to compute an approximation for $\mathbf{s}$ given $\mathbf{A}$ and $\mathbf{b}$.

Given the ill-posed nature of the problem from both the ill-conditioned $\mathbf{A}$ and the additive noise $\boldsymbol{\epsilon}$ in observation $\mathbf{d}$, a small error in $\mathbf{d}$ could lead to large deviations in the computed solution from the desired solution $\mathbf{s}$. Thus, a regularization technique is required to obtain a meaningful recovery or approximation to the true solution. The Bayesian framework provides an effective approach for this type of problems, as it naturally incorporates regularization through prior knowledge, meanwhile, quantifies the uncertainty in the desired parameters $\mathbf{s}$ utilizing the posterior density function [30, 9, 3].

[†]Emory University (ybu4@emory.edu)

[‡]Department of Mathematics, Emory University (jmchung@emory.edu).

Following the Bayesian approach, the solution of the inverse problem could be represented by the probability distribution of $\mathbf{s}$ given $\mathbf{d}$, denoted as $\mathbf{s}|\mathbf{d}$. Assume $\mathbf{s} \sim \mathcal{N}(\boldsymbol{\mu}, \lambda^{-2}\mathbf{Q})$ as the prior of which $\mathbf{s}$ is a Gaussian random variable with mean $\boldsymbol{\mu}$ and SPD covariance matrix $\mathbf{Q}$ and λ^2 is a scaling parameter for the precision matrix. Therefore, by Bayes' Theorem, the posterior probability distribution function is

$$p(\mathbf{s}|\mathbf{d}) \propto p(\mathbf{d}|\mathbf{s})p(\mathbf{s}) = \exp\left(-\frac{1}{2}\|\mathbf{A}\mathbf{s}-\mathbf{d}\|^2_{\mathbf{R}^{-1}} - \frac{\lambda^2}{2}\|\mathbf{s}-\boldsymbol{\mu}\|^2_{\mathbf{Q}^{-1}}\right) \tag{1.2}$$

where $\|\mathbf{x}\|_{\mathbf{M}} = \sqrt{\mathbf{x}^\top \mathbf{M}\mathbf{x}}$ is a vector norm for any SPD matrix $\mathbf{M}$. Similar to the maximum likelihood estimator, the maximum a posteriori (MAP) estimate provides a solution to (1.1) and can be obtained by minimizing the negative logarithm of the posterior

$$\begin{aligned} \mathbf{s}_\lambda &= \arg\min_{\mathbf{s}} -\log p(\mathbf{s}|\mathbf{d}) \\ &= \arg\min_{\mathbf{s}} \frac{1}{2}\|\mathbf{A}\mathbf{s}-\mathbf{d}\|^2_{\mathbf{R}^{-1}} + \frac{\lambda^2}{2}\|\mathbf{s}-\mu\|^2_{\mathbf{Q}^{-1}}. \end{aligned} \tag{1.3}$$

In fact, $\mathbf{s}_\lambda$ is the solution of a general-form Tikhonov problem, for which many approaches have been developed to compute a reliable solution through hybrid iterative methods [9, 24, 19, 13]. However, in this work, we further account for the uncertainties in the forward operator, by assuming that exact MVPs with $\mathbf{A}$ may not be available. Although such assumptions were considered for inexact Krylov methods [27], they have not been considered in previous methods for solving generalized Tikhonov problems.

This work addresses the scenario when the forward operator $\mathbf{A}$ is not fully known. This is a common issue in signal and image processing tasks [12], such as device calibration [6, 15], blind deconvolution [18], and super-resolution [5]. In these cases, the MVPs with $\mathbf{A}$ and $\mathbf{A}^\top$ cannot be performed exactly, i.e. they are only available as approximations denoted as $\widehat{\mathbf{A}}$ and $\widetilde{\mathbf{A}}^\top$.

Although the errors in $\mathbf{A}$ can be interpreted as model error, and there is prior work on statistical approaches to handle model error uncertainty [28], interpreting such model errors as random variables in a Bayesian framework can be computationally infeasible for our problems of interest. Another approach is to parameterize the forward model, hence reducing the number of unknown parameters defining the forward model [6]. However, computational methods that can incorporate uncertainty arising from the forward model focus mainly on computing point estimates, rather than uncertainty. Here, we assess the forward operator's inexactness through error matrices instead of as random variables or parameterized models. The MVPs with $\mathbf{A}$ and $\mathbf{A}^\top$ are available as approximations denoted as $\widehat{\mathbf{A}}$ and $\widetilde{\mathbf{A}}^\top$ and

$$\begin{aligned} \widehat{\mathbf{A}}_k\mathbf{x} &= (\mathbf{A}+\mathbf{E}_k)\mathbf{x} \\ \widetilde{\mathbf{A}}_k^\top\mathbf{y} &= (\mathbf{A}+\mathbf{F}_k)^\top\mathbf{y} \end{aligned} \tag{1.4}$$

where $\mathbf{E}_k$ and $\mathbf{F}_k$ are error matrices occurred at the k-th iteration. The primary reason for this approach is due to its large scale ($\mathbf{A} \in \mathbb{R}^{m\times n}$), which leads to the creation of too many unknown parameters and becomes computationally intractable. Moreover, our approach can handle scenarios where the prior covariance matrix is too large for obtaining a factorization.

Main contributions. In this paper, we propose the inexact generalized Golub-Kahan hybrid method to compute a solution for (1.3) where $\mathbf{A}$ contains errors. In particular, after an appropriate change of variables, the inexact generalized Golub-Kahan bidiagonalization methods is used to address inexactness in the forward model, meanwhile maintaining general to a rich class of covariance kernels. Further, a hybrid approach is adopted to automatically select a regularization parameter in solving the projected problem at each iteration. Numerical experiments show that the inexact generalized hybrid method can achieve a solution with high accuracy that is comparable to its exact counterpart.

The paper is organized as follows. In Section 2, we reformulate the problem through the change of variables to make it computationally feasible with the choice of covariance kernel $\mathbf{Q}$. Section 3 reviews the existing related iterative methods: Golub-Kahan Bidiagonalization for solving least squares (LS) problems, inexact Golub-Kahan decomposition for solving LS problems with inexact $\mathbf{A}$ and $\mathbf{A}^\top$, and generalized Golub-Kahan Bidiagonalization for solving generalized LS problems. Section 4 introduces the proposed inexact generalized Golub-Kahan method which is further adapted to the hybrid scheme in solving the Tikhonov regularized problem. Section 5 presents various numerical experiments, and Section 6 concludes this paper with some remarks and future directions.

2. General-form Tikhonov.

Our goal is to compute the MAP estimate (1.3) derived from the Bayesian approach. Note that the above MAP estimate can be written as the equivalent general-form Tikhonov problem,

$$\min_{\mathbf{s}} \frac{1}{2} \|\mathbf{L}_{\mathbf{R}}(\mathbf{A}\mathbf{s} - \mathbf{d})\|_2^2 + \frac{\lambda^2}{2} \|\mathbf{L}_{\mathbf{Q}}(\mathbf{s} - \boldsymbol{\mu})\|_2^2 \tag{2.1}$$

where $\mathbf{R}^{-1} = \mathbf{L}_{\mathbf{R}}{}^\top \mathbf{L}_{\mathbf{R}}$ and $\mathbf{Q}^{-1} = \mathbf{L}_{\mathbf{Q}}{}^\top \mathbf{L}_{\mathbf{Q}}$ are any symmetric matrix factorization (e.g., Cholesky). Iterative methods have been developed to solve (2.1), but these require computations with $\mathbf{L}_{\mathbf{Q}}$ and $\mathbf{L}_{\mathbf{R}}$ [2, 3, 17].

In this paper, we consider prior covariance matrices $\mathbf{Q}$ that are defined using kernel functions. More specifically, we choose the prior covariance kernel to be from the Matérn family. This choice allows for varying levels of smoothness and correlation structures between points. The entries of the covariance matrix are computed as $\mathbf{Q}_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j)$, where $\{\mathbf{x}\}_{i=1}^n$ are spatial points in the domain and the covariance kernel is given by

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = C_{\ell,\nu}(r) = \frac{1}{2^{\nu-1}\Gamma(\nu)} \left(\frac{\sqrt{2\nu} r}{\ell} \right)^{\nu} K_\nu \left(\frac{\sqrt{2\nu} r}{\ell} \right) \tag{2.2}$$

where $r = \|\mathbf{x}_i - \mathbf{x}_j\|_2$, Γ is the Gamma function, $K_\nu(\cdot)$ is the modified Bessel function of the second kind, ν and ℓ are positive parameters representing smoothness of the function and correlation length respectively. When $\nu = 1/2$, $C_{\ell,\nu}$ corresponds to the exponential covariance function, and when $\nu \to \infty$, $C_{\ell,\nu}$ converges to the Gaussian covariance function.

The Matérn covariance matrix is dense and thus the storage requirement and the computational cost for a matrix-vector product are $\mathcal{O}(n^2)$, which can be prohibitively expensive. However, by exploiting the stationary or invariant property of the Matérn family of covariance kernels and the regular equispaced spatial grid, the cost per matrix-vector product can

be reduced to $\mathcal{O}(n \log n)$ by utilizing methods such as the fast Fourier transform (FFT) or the fast multipole method [21]. This project uses the connection between the FFT and the Toeplitz structure for efficient computation of $\mathbf{Qx}$.

With our interest in incorporating Matérn covariance matrices, the computation of $\mathbf{Q}^{-1}$, $\mathbf{L_Q}^\top$, or $\mathbf{L_Q}$ could be very expensive. Thus, to avoid such computations while directly solving the Tikhonov normal equation or through priorconditioning, a change of variables

$$\mathbf{x} \leftarrow \mathbf{Q}^{-1}(\mathbf{s} - \boldsymbol{\mu}), \quad \mathbf{b} \leftarrow \mathbf{d} - \mathbf{A}\boldsymbol{\mu}$$

could be applied [9].

Then, we obtain an equivalent problem

$$\mathbf{x}_\lambda = \arg\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{AQx} - \mathbf{b}\|^2_{\mathbf{R}^{-1}} + \frac{\lambda^2}{2} \|\mathbf{x}\|^2_{\mathbf{Q}} \tag{2.3}$$

with the MAP estimate $\mathbf{s}_\lambda = \boldsymbol{\mu} + \mathbf{Q}\mathbf{x}_\lambda$ and the new Bayesian interpretation

$$\mathbf{b}|\mathbf{x} \sim \mathcal{N}(\mathbf{AQx}, \mathbf{R^{-1}}), \quad \mathbf{x} \sim \mathcal{N}(\mathbf{0}, \lambda^{-2}\mathbf{Q^{-1}}).$$

Thus, the MAP estimate could be reformulated as a LS problem as in (2.3), with $\mathbf{x} = \mathbf{Q}^{-1}(\mathbf{s} - \boldsymbol{\mu})$, $\mathbf{b} = \mathbf{d} - \mathbf{A}\boldsymbol{\mu}$, $\mathbf{A}$ being a known forward process, and $\mathbf{Q}$ is a covariance matrix from the Matérn family. $\mathbf{A}$ and $\mathbf{Q}$ are really large so they cannot be explicitly stored and are only accessible through MVPs as function handles.

3. Background on Iterative Methods for Inverse Problems.

The goal of this section is to introduce several established iterative methods in solving related problems, upon which the proposed inexact gen-GK bidiagonalization detailed in Section 4 is built upon.

3.1. Golub-Kahan Bidiagonalization.

Given an unregularized standard LS problem,

$$\min_{\mathbf{x} \in \mathbb{R}^n} \|\mathbf{Ax} - \mathbf{b}\|_2^2, \tag{3.1}$$

the Golub-Kahan (GK) bidiagonalization is one of the most common iterative methods which project the stated problem onto subspaces of increasing dimension (e.g. Krylov subspaces) [14, 4]. It generates two sets of orthogonal vectors to span the Krylov subspaces $\mathcal{K}_k(\mathbf{AA}^\top, \mathbf{b})$ and $\mathcal{K}_k(\mathbf{A}^\top\mathbf{A}, \mathbf{A}^\top\mathbf{b})$.

Given $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $\mathbf{b} \in \mathbb{R}^m$, the GK process initiates $\mathbf{u}_1^{\mathrm{s}} = \mathbf{b}/\beta_1^{\mathrm{s}}$ where $\beta_1^{\mathrm{s}} = \|\mathbf{b}\|_2$ and $\mathbf{v}_1^{\mathrm{s}} = \mathbf{A}^\top\mathbf{u}_1^{\mathrm{s}}/\alpha_1^{\mathrm{s}}$ where $\alpha_1^{\mathrm{s}} = \left\|\mathbf{A}^\top\mathbf{u}_1^{\mathrm{s}}\right\|_2$. The superscript 's' denotes the basis generated for standard GK bidiagonalization. Then, at the k-th iteration, it method generates vectors $\mathbf{u}_{k+1}^{\mathrm{s}}$ and $\mathbf{v}_{k+1}^{\mathrm{s}}$ and scalars $\alpha_{k+1}^{\mathrm{s}}$ and β_{k+1}^{s}, as diagonal and sub-diagonal entries in $\mathbf{B}_k^{\mathrm{s}}$, through

$$\mathbf{u}_{k+1}^{\mathrm{s}} = (\mathbf{A}\mathbf{v}_k^{\mathrm{s}} - \alpha_k^{\mathrm{s}}\mathbf{u}_k^{\mathrm{s}})/\beta_{k+1}^{\mathrm{s}}, \qquad \mathbf{v}_{k+1}^{\mathrm{s}} = (\mathbf{A}^\top\mathbf{u}_{k+1}^{\mathrm{s}} - \beta_{k+1}^{\mathrm{s}}\mathbf{v}_k^{\mathrm{s}})/\alpha_{k+1}^{\mathrm{s}}, \tag{3.2}$$

where the values of $\alpha_{k+1}^{\mathrm{s}}$ and β_{k+1}^{s} are chosen to ensure $\left\|\mathbf{v}_{k+1}^{\mathrm{s}}\right\|_2 = 1$ and $\left\|\mathbf{u}_{k+1}^{\mathrm{s}}\right\|_2 = 1$ respectively.

After k iterations of the GK process, we obtain

$$\mathbf{B}_k^{\mathrm{s}} = \begin{bmatrix} \alpha_1^{\mathrm{s}} & 0 & \dots & 0 \\ \beta_2^{\mathrm{s}} & \alpha_2^{\mathrm{s}} & \ddots & \vdots \\ 0 & \beta_3^{\mathrm{s}} & \ddots & 0 \\ \vdots & \ddots & \ddots & \alpha_k^{\mathrm{s}} \\ 0 & \dots & 0 & \beta_{k+1}^{\mathrm{s}} \end{bmatrix} \in \mathbb{R}^{(k+1)\times k}, \quad \mathbf{U}_{k+1}^{\mathrm{s}} = [\mathbf{u}^{\mathrm{s}}, \dots, \mathbf{u}_{k+1}^{\mathrm{s}}], \quad \mathbf{V}_k^{\mathrm{s}} = [\mathbf{v}_1^{\mathrm{s}}, \dots, \mathbf{v}_k^{\mathrm{s}}],$$

with following relationships

$$\begin{aligned} \mathbf{U}_{k+1}^{\mathrm{s}} \beta_1^{\mathrm{s}} \mathbf{e}_1 &= \mathbf{b}, \\ \mathbf{A}\mathbf{V}_k^{\mathrm{s}} &= \mathbf{U}_{k+1}^{\mathrm{s}} \mathbf{B}_k^{\mathrm{s}}, \\ \mathbf{A}^\top \mathbf{U}_{k+1}^{\mathrm{s}} &= \mathbf{V}_k^{\mathrm{s}} (\mathbf{B}_k^{\mathrm{s}})^\top + \alpha_{k+1}^{\mathrm{s}} \mathbf{v}_{k+1}^{\mathrm{s}} \mathbf{e}_{k+1}^\top, \end{aligned} \tag{3.3}$$

where $\mathbf{e}_i$ is the ith column of the identity matrix of appropriate size.

LSQR is an iterative method for solving (3.1) where at the kth iteration, we seek a solution $\mathbf{x}_k^{\mathrm{s}} \in \mathrm{span}\{\mathbf{V}_k^{\mathrm{s}}\}$. Define the residual at k-th iteration as

$$\mathbf{r}_k \equiv \mathbf{A}\mathbf{x}_k^{\mathrm{s}} - \mathbf{b} = \mathbf{A}\mathbf{V}_k^{\mathrm{s}} \mathbf{y}_k^{\mathrm{s}} - \mathbf{b} = \mathbf{U}_{k+1}^{\mathrm{s}} (\mathbf{B}_k^{\mathrm{s}} \mathbf{y}_k^{\mathrm{s}} - \beta_1^{\mathrm{s}} \mathbf{e}_1),$$

where $\mathbf{y}_k^{\mathrm{s}} \in \mathbb{R}^k$. Then, we could solve for $\mathbf{y}_k^{\mathrm{s}}$ by solving the following projected LS problem

$$\mathbf{y}_k^{\mathrm{s}} = \min_{\mathbf{y}\in\mathbb{R}^k} \|\mathbf{B}_k^{\mathrm{s}} \mathbf{y} - \beta_1^{\mathrm{s}} \mathbf{e}_1\|_2^2, \tag{3.4}$$

and recover the solution as $\mathbf{x}_k^{\mathrm{s}} = \mathbf{V}_k^{\mathrm{s}} \mathbf{y}_k^{\mathrm{s}}$.

3.2. Inexact Golub-Kahan Decomposition. Consider the same linear system as in (3.1), but assume that the MVPs with $\mathbf{A}$ and $\mathbf{A}^\top$ are only approximately available, i.e. at the k-th iteration, we have approximate MVPs as defined in (1.4) In this scenario, the inexact Golub-Kahan (iGK) decomposition [12] provides an efficient method for computing a solution subspace.

The iteration-wise computation and matrix factorization of iGK are different from GK in that they adapt to the inexactness in MVPs. Similar to the notation used for GK, we use the superscripts i to represent the vectors and matrices computed using iGK. Initializing $\mathbf{u}_1^{\mathrm{i}} = \mathbf{b}/\beta^{\mathrm{i}}$ where $\beta^{\mathrm{i}} = \|\mathbf{b}\|_2$, and $\mathbf{v}_1^{\mathrm{i}} = (\mathbf{A} + \mathbf{F}_1)^\top \mathbf{u}_1^{\mathrm{i}} / [\mathbf{L}^{\mathrm{i}}]_{1,1}$ where $[\mathbf{L}^{\mathrm{i}}]_{1,1} = \left\|(\mathbf{A} + \mathbf{F}_1)^\top \mathbf{u}_1^{\mathrm{i}}\right\|_2$, the k-th iteration computes

$$\begin{aligned} \bar{\mathbf{u}}_k^{\mathrm{i}} &= (\mathbf{A} + \mathbf{E}_k)\mathbf{v}_k^{\mathrm{i}}, & \mathbf{u}^{\mathrm{i}} &= (\mathbf{I} - \mathbf{U}_k^{\mathrm{i}} (\mathbf{U}_k^{\mathrm{i}})^\top) \bar{\mathbf{u}}_k^{\mathrm{i}}, & \mathbf{u}_{k+1}^{\mathrm{i}} &= \mathbf{u}^{\mathrm{i}} / \left\|\mathbf{u}^{\mathrm{i}}\right\|_2, \\ \bar{\mathbf{v}}_{k+1}^{\mathrm{i}} &= (\mathbf{A} + \mathbf{F}_{k+1})^\top \mathbf{u}_{k+1}^{\mathrm{i}}, & \mathbf{v}^{\mathrm{i}} &= (\mathbf{I} - \mathbf{V}_k^{\mathrm{i}} (\mathbf{V}_k^{\mathrm{i}})^\top) \bar{\mathbf{v}}_{k+1}^{\mathrm{i}}, & \mathbf{v}_{k+1}^{\mathrm{i}} &= \mathbf{v}^{\mathrm{i}} / \left\|\mathbf{v}^{\mathrm{i}}\right\|_2, \end{aligned} \tag{3.5}$$

where $\mathbf{U}_k^{\mathrm{i}} = [\mathbf{u}_1^{\mathrm{i}}, \dots, \mathbf{u}_k^{\mathrm{i}}] \in \mathbb{R}^{m\times k}$ and $\mathbf{V}_k^{\mathrm{i}} = [\mathbf{v}_1^{\mathrm{i}}, \dots, \mathbf{v}_k^{\mathrm{i}}] \in \mathbb{R}^{n\times k}$ are matrices with orthonormal columns.

After k iterations, iGK algorithm computes upper Hessenberg matrix $\mathbf{M}_k^{\mathrm{i}} \in \mathbb{R}^{(k+1)\times k}$ with $[\mathbf{M}_k^{\mathrm{i}}]_{j,i} = (\mathbf{u}_j^{\mathrm{i}})^\top \bar{\mathbf{u}}_i^{\mathrm{i}}$ and $[\mathbf{M}_k^{\mathrm{i}}]_{i+1,i} = \left\|\mathbf{u}^{\mathrm{i}}\right\|_2$ for $1 \le j \le i \le k$, lower triangular matrix

$\mathbf{L}_{k+1}^{\mathrm{i}} \in \mathbb{R}^{(k+1)\times(k+1)}$ with $[\mathbf{L}_{k+1}^{\mathrm{i}}]_{i+1,j} = (\mathbf{v}_j^{\mathrm{i}})^\top \bar{\mathbf{v}}_{i+1}^{\mathrm{i}}$ and $[\mathbf{L}_{k+1}^{\mathrm{i}}]_{i+1,i+1} = \|\mathbf{v}_i^{\mathrm{i}}\|_2$ for $1 \le j \le i \le k$, with the following relationships

$$
\begin{aligned}
(\mathbf{A}+\mathcal{E}_k)\mathbf{V}_k^{\mathrm{i}} &= \mathbf{U}_{k+1}^{\mathrm{i}}\mathbf{M}_k^{\mathrm{i}}, & \mathcal{E}_k &= \sum_{i=1}^{k} \mathbf{E}_i \mathbf{v}_i^{\mathrm{i}} (\mathbf{v}_i^{\mathrm{i}})^\top, \\
(\mathbf{A}+\mathcal{F}_{k+1})^\top \mathbf{U}_{k+1}^{\mathrm{i}} &= \mathbf{V}_{k+1}^{\mathrm{i}} (\mathbf{L}_{k+1}^{\mathrm{i}})^\top, & \mathcal{F}_{k+1} &= \sum_{i=1}^{k+1} (\mathbf{u}_i^{\mathrm{i}} (\mathbf{u}_i^{\mathrm{i}})^\top) \mathbf{F}_i.
\end{aligned}
\tag{3.6}
$$

To ensure the orthogonality of $\mathbf{V}_k^{\mathrm{i}}$ and $\mathbf{U}_{k+1}^{\mathrm{i}}$ under the inexactness of $\mathbf{A}$ and $\mathbf{A}^\top$, iGK generates $\mathbf{M}_k^{\mathrm{i}}$ and $\mathbf{L}_{k+1}^{\mathrm{i}}$ instead of $\mathbf{B}_k^{\mathrm{s}}$ as in (3.3).

Here, the inexact LSQR (iLSQR) is used solve for (3.1) with inexact $\mathbf{A}$ and $\mathbf{A}^\top$. At the k-th iteration, it computes

$$
\mathbf{y}_k^{\mathrm{i}} = \arg\min_{\mathbf{y}\in\mathbb{R}^k} \left\| \mathbf{M}_k^{\mathrm{i}} \mathbf{y} - \beta^{\mathrm{i}} \mathbf{e}_1 \right\|_2^2 \tag{3.7}
$$

and obtain the solution $\mathbf{x}_k^{\mathrm{i}} = \mathbf{V}_k^{\mathrm{i}} \mathbf{y}_k^{\mathrm{i}}$. Note that due to inxactness in the MVPs, the k-th iteration of iLSQR does not minimize the exact residual norm $\|\mathbf{A}\mathbf{x}_k - \mathbf{b}\|_2$ in the subspace $\mathrm{span}\{\mathbf{V}_k^{\mathrm{i}}\}$. Moreover, $\mathrm{span}\{\mathbf{V}_k^{\mathrm{i}}\}$ and $\mathrm{span}\{\mathbf{U}_{k+1}^{\mathrm{i}}\}$ are no longer Krylov subspaces.

3.3. Generalized Golub-Kahan Bidiagonalization. Generalized Golub-Kahan (genGK) Bidiagonalization is an iterative method designed to solve linear systems in the generalized least-squares sense [1, 9]. Referring to the problem we have in (2.3) where exact MVPs with $\mathbf{A}$ and $\mathbf{A}^\top$ could be achieved, genGk generates two sets of orthogonal vectors that span the Krylov subspaces $\mathcal{K}_k(\mathbf{A}^\top\mathbf{R}^{-1}\mathbf{A}\mathbf{Q}, \mathbf{A}^\top\mathbf{R}^{-1}\mathbf{b})$ and $\mathcal{K}_k(\mathbf{A}\mathbf{Q}\mathbf{A}^\top\mathbf{R}^{-1}, \mathbf{b})$.

Given matrices $\mathbf{A}$, $\mathbf{R}$, $\mathbf{Q}$ and vector $\mathbf{b}$, the genGK process initiates $\mathbf{u}_1^{\mathrm{g}} = \mathbf{b}/\beta_1^{\mathrm{g}}$ where $\beta_1^{\mathrm{g}} = \|\mathbf{b}\|_{\mathbf{R}^{-1}}$, and $\alpha_1^{\mathrm{g}}\mathbf{v}_1^{\mathrm{g}} = \mathbf{A}^\top\mathbf{R}^{-1}\mathbf{u}_1^{\mathrm{g}}$ where $\alpha_1^{\mathrm{g}} = \|\mathbf{v}_1^{\mathrm{g}}\|_{\mathbf{Q}}$. We use the superscript 'g' to denote the vectors and matrices computed from the genGK process. At the k-th iteration, genGK generates $\mathbf{u}_{k+1}^{\mathrm{g}}$ and $\mathbf{v}_{k+1}^{\mathrm{g}}$ through

$$
\begin{aligned}
\beta_{k+1}^{\mathrm{g}}\mathbf{u}_{k+1}^{\mathrm{g}} &= \mathbf{A}\mathbf{Q}\mathbf{v}_{k+1}^{\mathrm{g}} - \alpha_k^{\mathrm{g}}\mathbf{u}_k^{\mathrm{g}}, \\
\alpha_{k+1}^{\mathrm{g}}\mathbf{v}_{k+1}^{\mathrm{g}} &= \mathbf{A}^\top\mathbf{R}^{-1}\mathbf{u}_{k+1}^{\mathrm{g}} - \beta_{k+1}^{\mathrm{g}}\mathbf{v}_k^{\mathrm{g}},
\end{aligned}
\tag{3.8}
$$

where scalars $\alpha_{k+1}^{\mathrm{g}}, \beta_{k+1}^{\mathrm{g}} \ge 0$ are chosen such that $\|\mathbf{u}_{k+1}^{\mathrm{g}}\|_{\mathbf{R}^{-1}} = \|\mathbf{v}_{k+1}^{\mathrm{g}}\|_{\mathbf{Q}} = 1$.

After k iterations, the algorithm generates

$$
\mathbf{B}_k^{\mathrm{g}} = \begin{bmatrix} \alpha_1^{\mathrm{g}} & 0 & \dots & 0 \\ \beta_2^{\mathrm{g}} & \alpha_2^{\mathrm{g}} & \ddots & \vdots \\ 0 & \beta_3^{\mathrm{g}} & \ddots & 0 \\ \vdots & \ddots & \ddots & \alpha_k^{\mathrm{g}} \\ 0 & \dots & 0 & \beta_{k+1}^{\mathrm{g}} \end{bmatrix}, \quad \mathbf{U}_{k+1}^{\mathrm{g}} = [\mathbf{u}^{\mathrm{g}}, \dots, \mathbf{u}_{k+1}^{\mathrm{g}}], \quad \mathbf{V}_k^{\mathrm{g}} = [\mathbf{v}_1^{\mathrm{g}}, \dots, \mathbf{v}_k^{\mathrm{g}}]
$$

with the following relations holding up to machine precision

$$\begin{aligned} \mathbf{U}_{k+1}^{\mathrm{g}}\beta_1^{\mathrm{g}}\mathbf{e}_1 &= \mathbf{b}, \\ \mathbf{A}\mathbf{Q}\mathbf{V}_k^{\mathrm{g}} &= \mathbf{U}_{k+1}^{\mathrm{g}}\mathbf{B}_k^{\mathrm{g}}, \\ \mathbf{A}^\top\mathbf{R}^{-1}\mathbf{U}_{k+1}^{\mathrm{g}} &= \mathbf{V}_k^{\mathrm{g}}(\mathbf{B}_k^{\mathrm{g}})^\top + \alpha_{k+1}^{\mathrm{g}}\mathbf{v}_{k+1}^{\mathrm{g}}\mathbf{e}_{k+1}^\top, \end{aligned} \tag{3.9}$$

and matrices $\mathbf{U}_{k+1}^{\mathrm{g}}$ and $\mathbf{V}_k^{\mathrm{g}}$ satisfy the following orthogonality conditions:

$$(\mathbf{V}_k^{\mathrm{g}})^\top\mathbf{Q}\mathbf{V}_k^{\mathrm{g}} = \mathbf{I}_k \quad \text{and} \quad (\mathbf{U}_{k+1}^{\mathrm{g}})^\top\mathbf{R}^{-1}\mathbf{U}_{k+1}^{\mathrm{g}} = \mathbf{I}_{k+1}. \tag{3.10}$$

Again, we can consider an iterative method where we seek a solution $\mathbf{x}_k^{\mathrm{g}}$ in the span of $\mathbf{V}_k^{\mathrm{g}}$, i.e., $\mathbf{x}_k^{\mathrm{g}} = \mathbf{V}_k^{\mathrm{g}}\mathbf{y}_k^{\mathrm{g}}$, where at each iteration, we solve for $\mathbf{y}_k^{\mathrm{g}}$

$$\mathbf{y}_k^{\mathrm{g}} = \min_{\mathbf{y}\in\mathbb{R}^n} \left\|\mathbf{B}_k^{\mathrm{g}}\mathbf{y} - \beta_1^{\mathrm{g}}\mathbf{e}_1\right\|_2^2 + \frac{\lambda^2}{2}\left\|\mathbf{y}\right\|_2^2. \tag{3.11}$$

We call this the genLSQR method.

4. Iterative methods based on inexact generalized Golub-Kahan decomposition. This section proposes an iterative solver based on the inexact generalized Golub-Kahan (igenGK) decomposition. This method aims to solve problems where matrices $\mathbf{A}$ and $\mathbf{Q}$ are really large and can only be accessed via function evaluations, for example the least squares problem as stated in (2.3). The main difference is that we allow potential inexactness in $\mathbf{A}$ and $\mathbf{A}^\top$.

4.1. Inexact generalized Golub-Kahan decomposition. Assume the MVPs with $\mathbf{A}$ and $\mathbf{A}^\top$ cannot be performed exactly, following the setup as in (1.4), and the covariance kernel matrix $\mathbf{Q}$ can only be accessed through MVPs. Here, we propose an inexact generalized Golub-Kahan decomposition method for solving such problem.

The igenGK decomposition method combines the iGK and genGK approaches. With initializations $\mathbf{u}_1 = \mathbf{b}/\beta$ where $\beta = \|\mathbf{b}\|_{\mathbf{R}^{-1}}$, $\mathbf{v}_1 = (\mathbf{A}+\mathbf{F}_1)^\top\mathbf{R}^{-1}\mathbf{u}_1/[\mathbf{L}]_{1,1}$ where $[\mathbf{L}]_{1,1} = \left\|(\mathbf{A}+\mathbf{F}_1)^\top\mathbf{R}^{-1}\mathbf{u}_1\right\|_{\mathbf{Q}}$, at the k-th iteration, it computes

$$\begin{aligned} \bar{\mathbf{u}}_k &= (\mathbf{A}+\mathbf{E}_k)\mathbf{Q}\mathbf{v}_k, & \mathbf{u} &= (\mathbf{I}-\mathbf{U}_k\mathbf{U}_k^\top\mathbf{R}^{-1})\bar{\mathbf{u}}_k, & \mathbf{u}_{k+1} &= \mathbf{u}/\left\|\mathbf{u}\right\|_{\mathbf{R}^{-1}}, \\ \bar{\mathbf{v}}_{k+1} &= (\mathbf{A}+\mathbf{F}_{k+1})^\top\mathbf{R}^{-1}\mathbf{u}_{k+1}, & \mathbf{v} &= (\mathbf{I}-\mathbf{V}_k\mathbf{V}_k^\top\mathbf{Q})\bar{\mathbf{v}}_{k+1}, & \mathbf{v}_{k+1} &= \mathbf{v}/\left\|\mathbf{v}\right\|_{\mathbf{Q}}, \end{aligned}$$

where $\mathbf{U}_{k+1} = [\mathbf{u}_1,\ldots,\mathbf{u}_{k+1}] \in \mathbb{R}^{m\times(k+1)}$ and $\mathbf{V}_k = [\mathbf{v}_1,\ldots,\mathbf{v}_k] \in \mathbb{R}^{n\times k}$ are matrices with orthonormal columns, separately with respect to $\mathbf{R}^{-1}$ and $\mathbf{Q}$:

$$\mathbf{U}_{k+1}^\top\mathbf{R}^{-1}\mathbf{U}_{k+1} = \mathbf{I}_{k+1} \quad \text{and} \quad \mathbf{V}_k^\top\mathbf{Q}\mathbf{V}_k = \mathbf{I}_k. \tag{4.1}$$

Thus, after k iterations, it generates an upper Hessenberg matrix $\mathbf{M}_k \in \mathbb{R}^{(k+1)\times k}$ with $[\mathbf{M}_k]_{j,i} = \mathbf{u}_j^\top\mathbf{R}^{-1}\bar{\mathbf{u}}_i$ and $[\mathbf{M}_k]_{i+1,i} = \|\mathbf{u}\|_{\mathbf{R}^{-1}}$ for $1 \le j \le i \le k$; a lower triangular matrix $\mathbf{L}_{k+1} \in \mathbb{R}^{(k+1)\times(k+1)}$ with $[\mathbf{L}_{k+1}]_{i+1,j} = \mathbf{v}_j^\top\mathbf{Q}\bar{\mathbf{v}}_{i+1}$ and $[\mathbf{L}_{k+1}]_{i+1,i+1} = \|\mathbf{v}\|_{\mathbf{Q}}$ for $1 \le j \le i \le k$; as well as following relationships

$$\begin{aligned} (\mathbf{A}+\mathcal{E}_k)\mathbf{Q}\mathbf{V}_k &= \mathbf{U}_{k+1}\mathbf{M}_k, \\ (\mathbf{A}+\mathcal{F}_{k+1})^\top\mathbf{R}^{-1}\mathbf{U}_{k+1} &= \mathbf{V}_{k+1}\mathbf{L}_{k+1}^\top, \end{aligned} \tag{4.2}$$

where $\mathcal{E}_k = \sum_{i=1}^{k} \mathbf{E}_i \mathbf{Q}\mathbf{v}_i\mathbf{v}_i^\top$, $\mathcal{F}_k = \sum_{i=1}^{k} (\mathbf{u}_i\mathbf{u}_i^\top)\mathbf{R}^{-1}\mathbf{F}_i$. To see this, notice that

$$\begin{aligned}[(\mathbf{A}+\mathbf{E}_1)\mathbf{Q}\mathbf{v}_1, \ldots, (\mathbf{A}+\mathbf{E}_k)\mathbf{Q}\mathbf{v}_k] =& \mathbf{A}\mathbf{Q}\mathbf{V}_k + [\mathbf{E}_1\mathbf{Q}\mathbf{v}_1, \ldots, \mathbf{E}_k\mathbf{Q}\mathbf{v}_k]\mathbf{V}_k^\top\mathbf{Q}\mathbf{V}_k \\ =& \mathbf{A}\mathbf{Q}\mathbf{V}_k + \left(\sum_{i=1}^{k} \mathbf{E}_i\mathbf{Q}\mathbf{v}_i\mathbf{v}_i^\top\right)\mathbf{Q}\mathbf{V}_k \\ =& (\mathbf{A}+\mathcal{E}_k)\mathbf{Q}\mathbf{V}_k,\end{aligned}$$

and similarly,

$$[(\mathbf{A}+\mathbf{F}_1)^\top\mathbf{R}^{-1}\mathbf{u}_1, \ldots, (\mathbf{A}+\mathbf{F}_k)^\top\mathbf{R}^{-1}\mathbf{u}_k] = (\mathbf{A}+\mathcal{F}_k)^\top\mathbf{R}^{-1}\mathbf{U}_k.$$

The following algorithm summarizes one iteration of the inexact generalized Golub-Kahan decomposition process. Given the inexactness of the forward matrix $\mathbf{A}$, one iteration of the algorithm performs $\widehat{\mathbf{A}}_k\mathbf{x}$ and $\widetilde{\mathbf{A}}_k^\top\mathbf{y}$ for the k-th iteration MVP. The inputs are $\mathbf{V}_{k-1} \in \mathbb{R}^{n\times(k-1)}$ with $\mathbf{V}_0$ as an empty matrix, $\mathbf{U}_k \in \mathbb{R}^{m\times k}$ with $\mathbf{U}_1 = \mathbf{b}/\left\|\mathbf{b}\right\|_{\mathbf{R}^{-1}}$, $\mathbf{M}_{k-1} \in \mathbb{R}^{k\times(k-1)}$ an upper Hessenberg matrix, and $\mathbf{C}_k = \mathbf{L}_k^\top \in \mathbb{R}^{k\times k}$ an upper triangular matrix.

Algorithm 4.1 One iteration of igenGK decomposition with Reorthogonalization

Input: $\widetilde{\mathbf{A}}_k, \widehat{\mathbf{A}}_k, \mathbf{Q}, \mathbf{R}, \mathbf{V}_{k-1}, \mathbf{U}_k, \mathbf{M}_{k-1}, \mathbf{C}_k$
if $k = 1$ **then**
  $\mathbf{v} = \widetilde{\mathbf{A}}_k^\top\mathbf{R}^{-1}\mathbf{u}_k$
  $\mathbf{v}_k = \mathbf{v}/c_{kk}$ where $c_{kk} = \left\|\mathbf{v}\right\|_{\mathbf{Q}}$
else
  $\mathbf{v} = \widetilde{\mathbf{A}}_k^\top\mathbf{R}^{-1}\mathbf{u}_k$
  for $j = 1, \ldots, k-1$ **do**
    $\mathbf{v} = \mathbf{v} - c_{ik}\mathbf{v}_j$ where $c_{ik} = \mathbf{v}_j^\top\mathbf{Q}\mathbf{v}$
  end for
  $\mathbf{v}_k = \mathbf{v}/c_{kk}$ where $c_{kk} = \left\|\mathbf{v}\right\|_{\mathbf{Q}}$
  $\mathbf{u} = \widehat{\mathbf{A}}_k\mathbf{Q}\mathbf{v}_k$
  for $j = 1, \ldots, k$ **do**
    $\mathbf{u} = \mathbf{u} - m_{jk}\mathbf{u_j}$ where $m_{jk} = \mathbf{u}_j^\top\mathbf{R}^{-1}\mathbf{u}$
  end for
  $\mathbf{u}_{k+1} = \mathbf{u}/m_{k+1,k}$ where $m_{k+1,k} = \left\|\mathbf{u}\right\|_{\mathbf{R}^{-1}}$
end if
return $\mathbf{V}_k = [\mathbf{V}_{k-1}\ \mathbf{v}_k], \mathbf{U}_{k+1} = [\mathbf{U}_k\ \mathbf{u}_{k+1}], \mathbf{M}_{k+1}, \mathbf{C}_k$

4.2. Solving the LS problem. In the subsection above, we introduce the igenGK process as an iterative method to generate a subspace for the solution. Further, we would like to solve the least-squares problem (2.3) through a sequence of projected LS problems which we denote the inexact generalized LSQR (igenLSQR) method. In particular, at the k-th step, we seek solution $\mathbf{x}_k \in \text{span}\{\mathbf{V}_k\}$, and define residual $\mathbf{r}_k \equiv (\mathbf{A}+\mathcal{E}_k)\mathbf{Q}\mathbf{x}_k - \mathbf{b}$, then we obtain the

following igenLSQR problem,

$$\min_{\mathbf{x}_k \in \text{span}\{\mathbf{V}_k\}} \frac{1}{2} \|(\mathbf{A} + \mathcal{E}_k)\mathbf{Q}\mathbf{x}_k - \mathbf{b}\|^2_{\mathbf{R}^{-1}} + \frac{\lambda^2}{2} \|\mathbf{x}_k\|^2_{\mathbf{Q}} . \tag{4.3}$$

For now, we may assume λ is fixed. Given $\mathbf{x}_k = \mathbf{V}_k\mathbf{y}_k$ where $\mathbf{y}_k \in \mathbb{R}^k$, we have

$$(\mathbf{A} + \mathcal{E}_k)\mathbf{Q}\mathbf{V}_k\mathbf{y}_k - \mathbf{b} = \mathbf{U}_{k+1}\mathbf{M}_k\mathbf{y}_k - \mathbf{b} = \mathbf{U}_{k+1}(\mathbf{M}_k\mathbf{y}_k - \beta\mathbf{e}_1)$$

and

$$\mathbf{y}_k^\top \mathbf{V}_k^\top \mathbf{Q}\mathbf{V}_k\mathbf{y}_k = \mathbf{y}_k^\top \mathbf{y}_k.$$

The above LS problem (4.3) could be formulated as follows

$$\min_{\mathbf{y}_k \in \mathbb{R}^k} \frac{1}{2} \|(\mathbf{A} + \mathcal{E}_k)\mathbf{Q}\mathbf{V}_k\mathbf{y}_k - \mathbf{b}\|^2_{\mathbf{R}^{-1}} + \frac{\lambda^2}{2} \|\mathbf{V}_k\mathbf{y}_k\|^2_{\mathbf{Q}} \tag{4.4}$$

which is equivalent to

$$\min_{\mathbf{y}_k \in \mathbb{R}^k} \frac{1}{2} \|\mathbf{M}_k\mathbf{y}_k - \beta\mathbf{e}_1\|^2_2 + \frac{\lambda^2}{2} \|\mathbf{y}_k\|^2_2 . \tag{4.5}$$

This approach is stemming from the standard LSQR [22, 23] and generalized LSQR [9].

After obtaining a solution for the projected problem, the solution $\mathbf{s}_k$ for the original problem (1.3) could be recovered as

$$\mathbf{s}_k = \boldsymbol{\mu} + \mathbf{Q}\mathbf{x}_k = \boldsymbol{\mu} + \mathbf{Q}\mathbf{V}_k\mathbf{y}_k. \tag{4.6}$$

4.3. Inexact generalized hybrid approach. So far, we have assumed the Tikhonov regularization parameter λ is known a priori. However, in practice, obtaining a good regularization parameter is crucial but could also be difficult, especially when the problem is large in scale. If a poor λ is chosen, it may lead to an imbalance between the regularization and perturbation error, thus ending up with poor solutions. In particular, a small λ can lead to overfitting and the solution being highly sensitive to additive noise, while a large λ would cause the solution to be dominated by regularization error. In this section, we propose an inexact generalized hybrid approach, where the regularization parameter can be automatically estimated at each iteration.

This method follows from many previous works, where the problem is first projected down to a lower dimensional space and the projected problem (4.5) would be further solved through various parameter selection methods [16, 17, 4].

While various regularization parameter methods are available [7, 8, 20, 25], here we consider the discrepancy principle (DP) and the weighted generalized cross validation (WGCV) approach. To provide a benchmark for parameter selection method comparisons, both will be compared with the optimal approach, where the regularization parameter λ_{opt} is chosen to minimize the 2-norm of the error between the reconstructed solution and the true solution

$$\lambda_{\text{opt}} = \arg\min_{\lambda} \|\mathbf{s}_k(\lambda) - \mathbf{s}_{\text{true}}\|^2_2 , \tag{4.7}$$

where $\mathbf{s}_{\text{true}}$ denotes the true solution and $\mathbf{s}_k(\lambda)$ denotes the solution computed at the k-th iteration using regularization parameter λ.

At the k-th iteration, the exact residual is not available due to the inexactness arising from MVPs. Instead, we define the residual norm

$$\|\mathbf{r}_k(\lambda)\|_2 = \|(\mathbf{A} + \mathcal{E}_k)\mathbf{s}_k(\lambda) - \mathbf{d}\|_2 = \|\mathbf{M}_k\mathbf{y}_k(\lambda) - \beta_1\mathbf{e}_1\|_2.$$

The discrepancy principle (DP) method chooses $\lambda = \lambda_{\text{DP}}$ such that

$$\lambda_{\text{DP}} = \arg\min_{\lambda} \left| \|\mathbf{M}_k\mathbf{y}_k(\lambda) - \beta_1\mathbf{e}_1\|_2 - \nu_{\text{DP}}m \right| \tag{4.8}$$

where ν_{DP} is a user chosen constant and m is the expected value of $\|\boldsymbol{\epsilon}\|_{\mathbf{R}^{-1}}$.

The weighted generalized cross validation (WGCV) method is another common approach for selecting regularization parameters when the level of noise is unknown. This method follows from the statistical technique cross validation. By arbitrarily leaving out one element of the observed data $\mathbf{d}$, cross validation aims to find a good regularization parameter which is able to predict the missing element. However, these approaches can be expensive since they consider the full dimensional problem, so variants such as the generalized cross validation [14] method have been considered. Developed in the context of hybrid projection methods, the WGCV method selects the regularization parameter as

$$\lambda_{\text{WGCV}} = \arg\min_{\lambda} \frac{\|\mathbf{M}_k\mathbf{y}_k(\lambda) - \beta\mathbf{e}_1\|_2^2}{\text{trace}\left(\mathbf{I}_k - \omega\mathbf{M}_k(\mathbf{M}_k^\top\mathbf{M}_k + \lambda^2\mathbf{I})^{-1}\mathbf{M}_k\right)^2}, \tag{4.9}$$

where ω is a weight parameter that can be user defined o r e stimated d uring t he iterative process [7, 4].

5. Numerical Experiments. In this section, we consider two examples: the first o ne i s a X-ray computed tomographic (CT) reconstruction problem, presented from subsection 5.1 to subsection 5.4, and the second one is a problem in seismic travel-time tomography, presented in subsection 5.5. All experiments are performed in MATLAB R2023a, using the 'PRtomo' or 'PRseismic' example from the IRTools package [11]. Within each experiment, the proposed igenGK decomposition is compared with GK, iGK, and genGK with respect to their performance in image reconstruction. Also the relative reconstruction error norm at each iteration k is computed as

$$e_k = \|\mathbf{s}_k - \mathbf{s}_{\text{true}}\|_2 \,/\, \|\mathbf{s}_{\text{true}}\|_2 \,.$$

5.1. Numerical experiment 1: Comparison of iterative methods without regularization. We begin with the X-ray CT reconstruction problem, where the goal is to reconstruct the cross-section of an object from data collected along the X-rays penetrating the object. As shown in Figure 1, the X-ray source-detector pair rotates 180° around the object and a finite number of projections are collected at degrees $\boldsymbol{\theta}$, where $\boldsymbol{\theta}$ are projection angles and the resulting observation constitutes a 2D sinogram image which could be further vectorized as observation $\mathbf{d}$ in the inverse problem.

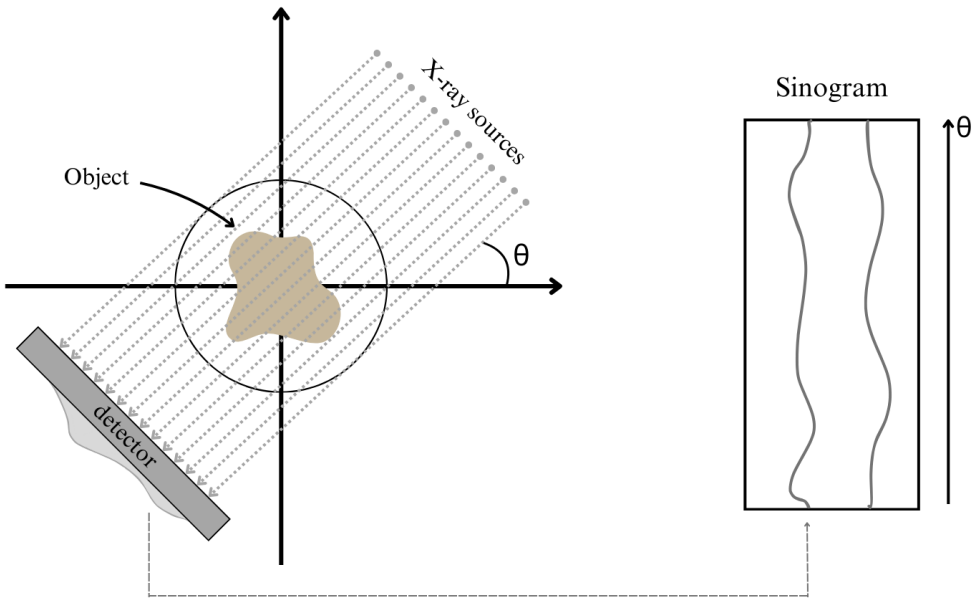


Figure 1: Set up of parallel-beam X-ray CT where the detector collects data from a finite number of projection angles $\boldsymbol{\theta}$ and the data from a set of $\boldsymbol{\theta}$ constitute the sinogram.

The first experiment compares the performance of GK, iGK, genGK, and igenGK methods without using any additional regularization on the projected problem (i.e., $\lambda = 0$). Note that in a Bayesian interpretation, this would correspond to a maximum likelihood estimate. However, in our iterative methods, $\mathbf{Q}$ is used to generate the solution subspace and has an impact on the solution, even for $\lambda = 0$.

PRtomo generates a 'medical' phantom image of size 128×128, so we stack the columns together to obtain $\mathbf{s}_{\text{true}} \in \mathbb{R}^{16384}$. The goal of this problem is to reconstruct the true image through the given forward X-ray CT process $\mathbf{A} \in \mathbb{R}^{6516\times 16384}$ and observation $\mathbf{d} \in \mathbb{R}^{6516}$. Suppose the matrix $\mathbf{A}$ is modeled after a parallel-beam process where the projections are recorded at every 5° with $\boldsymbol{\theta} = [1, 6, 11, \ldots, 176]$. Further, we assume the true image $\mathbf{s} \sim \mathcal{N}(\mathbf{0}, \lambda^{-2}\mathbf{Q})$ where Matérn covariance kernel $\mathbf{Q}$ is defined by $\nu = 1.5$, $\ell = 0.01$, and Gaussian white noise is added to $\mathbf{d}_{\text{true}}$ to make it more realistic, so $\mathbf{d} = \mathbf{d}_{\text{true}} + \boldsymbol{\epsilon}$ where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma\mathbf{I})$, σ is chosen such that $\|\boldsymbol{\epsilon}\|_2 / \|\mathbf{d}_{\text{true}}\|_2 = 0.04$ indicating 4% noise level.

To simulate errors in the forward model, we incorporate random inexactness within each MVP with $\mathbf{A}$ and $\mathbf{A}^\top$. To implement this, the forward matrix is defined as an object (i.e., using object-oriented programming) and is accessed through function evaluations for MVPs with $\mathbf{A}$ and $\mathbf{A}^\top$. More precisely, at each iteration k, $\widehat{\mathbf{A}}_k\mathbf{x} = (\mathbf{A}+\mathbf{E}_k)\mathbf{x}$ and $\widetilde{\mathbf{A}}_k^\top\mathbf{x} = (\mathbf{A}^\top+\mathbf{F}_k)\mathbf{x}$ and $\mathbf{E}_k, \mathbf{F}_k$ are matrices whose entries are random numbers generated i.i.d. from the Gaussian distribution $\mathcal{N}(\mathbf{0}, \beta)$ with $\beta > 0$.

Firstly, we would like to verify the following relationships for the igenGK decomposition:

$$\mathbf{A}^\top \mathbf{U}_k = \mathbf{V}_k \mathbf{L}_k^\top, \tag{5.1}$$

$$\mathbf{A}\mathbf{Q}\mathbf{V}_k = \mathbf{U}_{k+1}\mathbf{M}_k, \tag{5.2}$$

$$\mathbf{I}_k = \mathbf{V}_k^\top \mathbf{Q}\mathbf{V}_k \quad \text{and} \quad \mathbf{I}_{k+1} = \mathbf{U}_{k+1}^\top \mathbf{R}^{-1}\mathbf{U}_{k+1}. \tag{5.3}$$

In Table 1, we present the results for various degrees of inexactness. We consider $k = 50$ and the following values for β: $10^{-2}, 10^{-4}$, and 10^{-6}. Results indicate that the discrepancies between the left and right hand sides of (5.1) and (5.2) are proportional to the amount of random inexactness introduced to each MVP with $\mathbf{A}$ and $\mathbf{A}^\top$, while (5.3) holds up to machine precision.

igenGK Relations	Error		
	$\beta = 10^{-2}$	$\beta = 10^{-4}$	$\beta = 10^{-6}$
$\left\|\mathbf{A}^\top \mathbf{U}_k - \mathbf{V}_k \mathbf{L}_k^\top\right\|_F / \left\|\mathbf{A}^\top \mathbf{U}_k\right\|_F$	5.26e-02	5.26e-04	5.26e-06
$\left\|\mathbf{A}\mathbf{Q}\mathbf{V}_k - \mathbf{U}_{k+1}\mathbf{M}_k\right\|_F / \left\|\mathbf{A}\mathbf{Q}\mathbf{V}_k\right\|_F$	3.05e-02	3.07e-04	3.07e-06
$\left\|\mathbf{V}_k^\top \mathbf{Q}\mathbf{V}_k - \mathbf{I}_k\right\|_F / \sqrt{k}$	1.86e-15	2.63e-15	1.43e-15
$\left\|\mathbf{U}_{k+1}^\top \mathbf{R}^{-1}\mathbf{U}_{k+1} - \mathbf{I}_{k+1}\right\|_F / \sqrt{k+1}$	1.64e-14	1.03e-15	1.06e-14

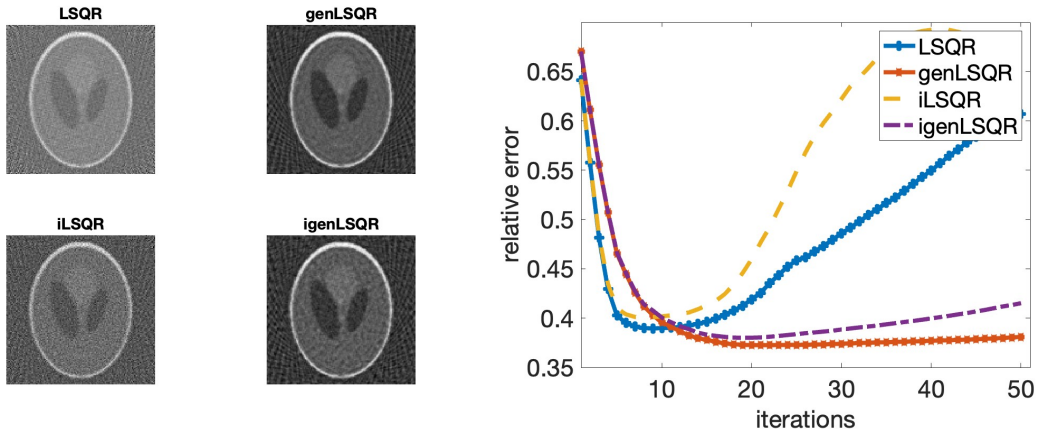
Table 1: Verifications of igenGK relationships for varying degrees of inexactness. Here, $\|\cdot\|_F$ denotes the Frobenius norm.

To further evaluate its capability in constructing the solution subspace, we evaluate the image reconstruction performance by directly solving $\mathbf{y}_k$ from (4.5) with $\lambda = 0$, i.e. the projected problems are solved without additional regularization technique. Then, we derive the estimate $\mathbf{s}_k$ in (1.3) by computing $\mathbf{s}_k = \mathbf{Q}\mathbf{x}_k = \mathbf{Q}\mathbf{V}_k\mathbf{y}_k$.

For comparative analysis, we further compute the estimates using the GK, iGK, and genGK iterative methods. For genGK and igenGK, which are generalized methods that require selecting a prior covariance matrix $\mathbf{Q}$, we set $\mathbf{Q}$ to be a kernel matrix defined by the Matérn kernel with $\nu = 1.5$ and $\ell = 0.01$. Both iGK and igenGK are inexact methods, and we use the inexact forward matrix $\mathbf{A}$ with $\beta = 10^{-2}$. The image reconstructions obtained from each method are presented in Figure 2a, while Figure 2b presents the relative error norms along iterations. The legend "LSQR" corresponds to standard GK iterations,"genLSQR" corresponds to the genGK iterative method,"iLSQR" corresponds to the inexact iterative method iGK, and "igenGK" correponds to inexact generalized iterative method igenGK.

We observe that, without additional regularization, all methods exhibit semiconvergence: they firstly converge in the first few iterations but after which the error increases dramatically when the inverted noise starts to dominate the solution [16]. So the reconstructed images at the end of 50 iterations are not desirable since they are contaminated with noise. From both Figure 2a and Figure 2b, while the solution from LSQR is close to iLSQR and genLSQR is close to igenLSQR, the generalized methods start with slower convergence but eventually perform better at later iterations with lower relative errors and less noisy images. Further, the

methods that involve inexactness (iLSQR and igenLSQR) perform worse but not significantly worse than their counterparts (LSQR and genLSQR), which is promising.



(a) Reconstruction

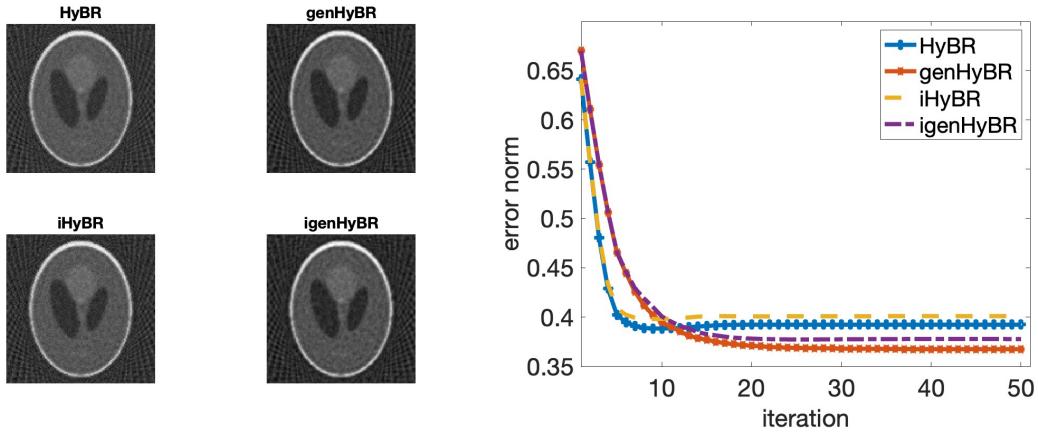
(b) Relative reconstruction error norms

Figure 2: Both (a) and (b) present comparisons of standard, generalized, inexact, and inexact generalized iterative methods for image reconstruction. The legend LSQR corresponds to the standard method, genLSQR corresponds to the generalized method, iLSQR cooresponds to the inexact method, and igenLSQR correponds to the inexact generalized method. In (a) we present the image reconstructions at the end of 50 iterations, and in (b) we provide the relative errors norms along the iterations.

5.2. Numerical experiment 2: Comparisons of hybrid methods with optimal regularization. As we observed in subsection 5.1, this large-scale problem without additional regularization on projected problems demonstrates semiconvergence. Thus, we seek the parameter-choice methods which automatically select the regularization parameters within the projected solution space at each iteration to achieve more stable and reliable solutions. In this experiment, following the same setup as the previous one, we further present the performance of the igenGK method when adopting a hybrid approach. Here, we consider the optimal regularization parameter $\lambda = \lambda_{\text{opt}}$ from (4.7).

Figure 3a and Figure 3b separately show the image reconstruction results and relative errors for all four iterative methods in adopting the hybrid approach. "HyBR" corresponds to standard hybrid method based on GK, "genHyBR" corresponds to the generalized hybrid method, "iHyBR" corresponds to the inexact hybrid method, and "igenHyBR" corresponds to the inexact generalized hybrid method. All of these results correspond to using the optimal regularization parameter.

We observe that the hybrid method not only leads to better reconstructed images but also stabilizes the convergence. Again, the igenHyBR we propose achieves a low relative reconstruction error, being only slightly higher than its exact counterpart.



(a) Reconstruction (b) Relative reconstruction error norms

Figure 3: Comparisons of standard, generalized, inexact, and inexact generalized hybrid approach with optimal regularization parameter. HyBR corresponds to the hybrid method based on GK, genHyBR corresponds to the generalized hybrid method based on genGK, iHyBR corresponds to the inexact hybrid method based on iGK, and igenHyBR corresponds to the inexact generalized hybrid method based on igenGK. In (a) we present the image reconstructions at the end of 50 iterations, (b) provides the relative errors norms along the iterations.

5.3. Numerical experiment 3: Comparison of regularization parameters. While the optimal regularization parameter is the parameter corresponding to the Tikhonov reconstruction that is closest to the true solution in the 2-norm sense, it is not realistic in applications since it requires the true solution which is not available. Therefore, we further compare two additional methods for choosing the regularization parameter, as introduced in subsection 4.3, that are more practical and commonly used.

In the following, we present an experiment to compare the different r egularization parameter selection methods concerning their efficacy in ap proximating th e tr ue so lution. We adopt the igenHyBR method with regularization parameters selected optimally, using the discrepancy principle (DP), and using the weighted generalized cross validation regularization (WGCV) method with ω selected adaptively at each iteration as described in [26]. We make comparisons of these three hybrid approaches. We see in Figure 4 that results for DP are similar to those for the optimal regularization parameter. This is very promising since the DP method does not require knowledge of true solution; however, it does require a noise level estimate. For these results, we use the actual noise level. Selecting regularization parameters with the WGCV method performs similarly to both the DP and the optimal regularization parameter methods in early iterations, achieving a low relative error. However, in later iterations, the error curve starts to increase for WGCV, indicating that the adaptive WGCV method does not consistently select a well-performing regularization parameter at every iteration. We observed similar results for the genHyBR method and additional tuning of the

weight parameter could improve the solutions.

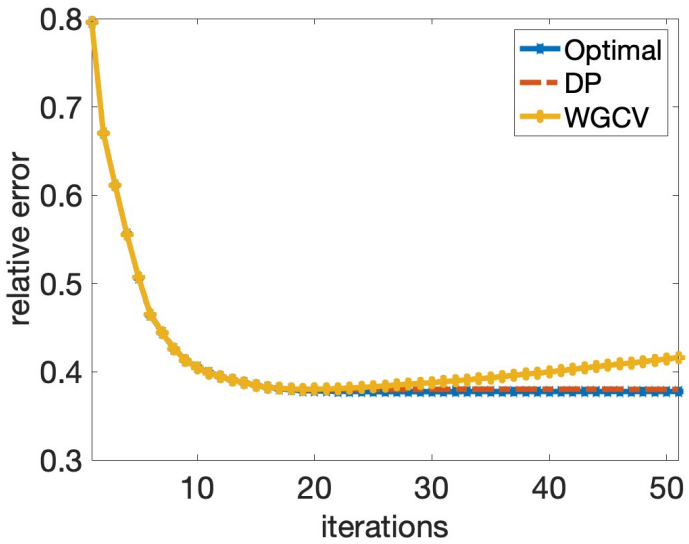


Figure 4: Comparison of different regularization parameter selection methods (optimal, DP, and WGCV) for choosing the Tikhonov regularization parameter λ within the igenHyBR framework. Relative reconstruction error norms are provided at each iteration.

We further investigate the choice of the weight parameter ω in the WGCV method by comparing the relative error curves computed with different choices of ω in Figure 5. To evaluate the effectiveness of the adaptive strategy, we compare it with fixed values $\omega = 1$, 0.95, 0.9, and 0.5. Both $\omega = 0.95$ and $\omega = 0.9$ outperform the adaptive approach with lower relative reconstruction errors, which suggests that a carefully chosen fixed ω can lead to better results in practice. However, if a good choice of ω is not known a priori, then the adaptive approach can give a reasonable estimate. Future work should be dedicated to finding a better ω in the adaptive WGCV method.

5.4. Numerical experiment 4: Inexactness in projection angles. In this experiment, we consider a more realistic scenario where the inexactness of the forward process arises from inaccurate projection angles. Continuing with the X-ray CT example, the forward process is determined by numerous factors, such as the experiment geometry (e.g., parallel or fan beam in 2D or cone beam in 3D), the projections angles (the degree angles chosen to conduct projections), and the number and spread of beams emitted from the X-ray sources at each projection angle. Uncertainty in any of these setups could lead to large variances in the observed sinogram and thus make the reconstructed image undesirable.

In reality, the degree at which the X-rays are taken and recorded is one of the main sources of uncertainty in CT reconstruction. Though we assume the exact projection angles are set before the scanning process, error arises due to inaccurate calibration of the equipment or unforeseen incidents, such as the involuntary movement of the scanned object, or the external

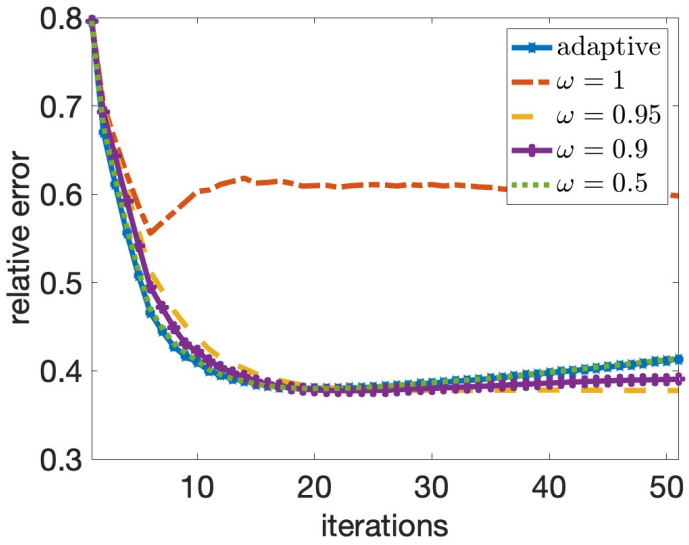


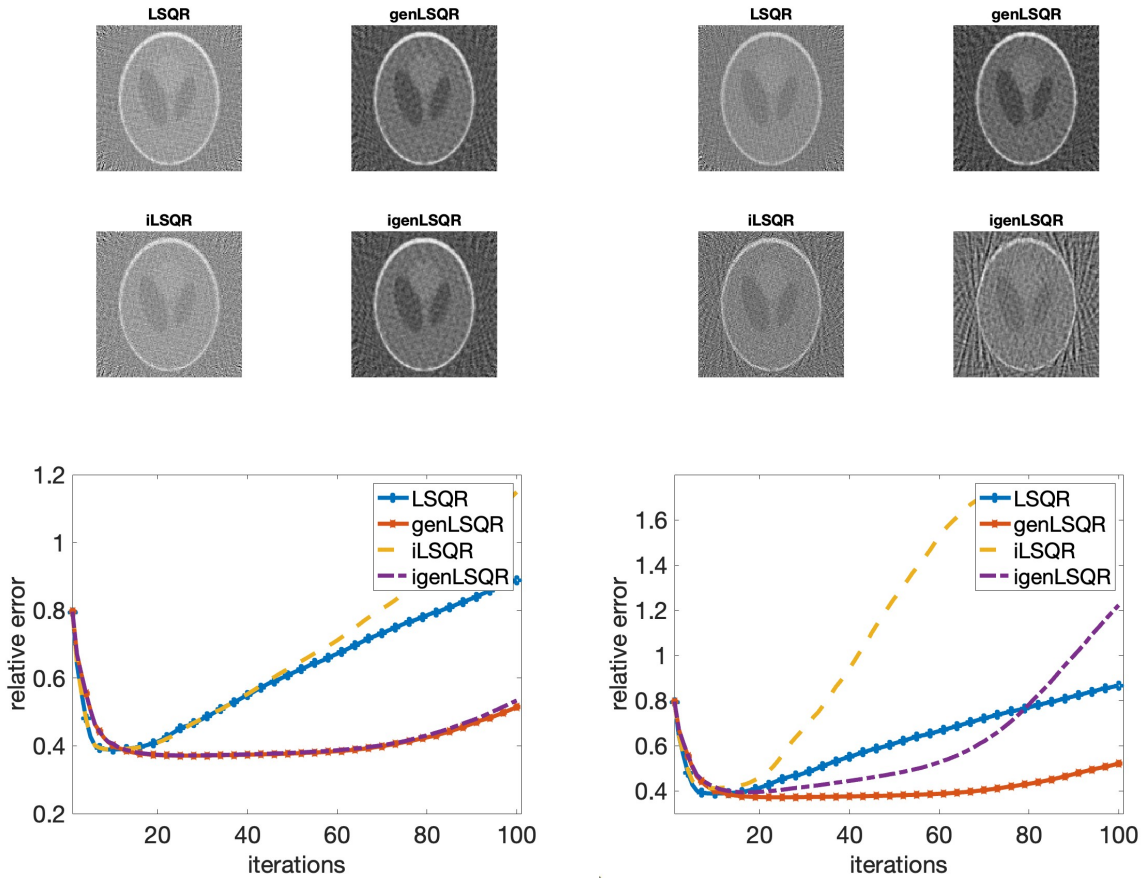
Figure 5: Comparison of WGCV parameter selection methods with different choices of ω.

factors causing vibrations in X-ray sources and detector [29].

The following example illustrates a scenario where the inexactness in the projection angles decreases over iterations. This is reasonable if one considers integrating these inexact methods within a larger optimization scheme where the angles are being updated and improved. That is, each solution could be adopted at the next iteration to better improve the estimated angles at which the X-rays are collected. Given the the accurate projection angles $\boldsymbol{\theta}_{\text{true}} = [1, 6, 11, \ldots, 176]^\top$, we simulate this scenario by using at each iteration k a set of inexact projection angles $\boldsymbol{\theta}_k = \boldsymbol{\theta}_{\text{true}} + \alpha_k \mathbf{e}_k$ where α_k decreases logarithmically along iterations and $\mathbf{e}_k$ is a random vector drawn from a standard normal distribution for each k.

In Figure 6, we present the image reconstruction and relative error results for two cases: the first case is presented in Figure 6a where we start with a small degree of inexactness $\alpha_1 = 10^{-1}$ and decreases logarithmically down to $\alpha_{100} = 10^{-6}$, and in the second case as shown in Figure 6b, we start from a larger inexactness $\alpha_1 = 10^{0}$ and similarly decreases down to 10^{-6}. Both of them are solved without additional regularization on the projected problems ($\lambda = 0$) and we extend the iteration number to 100 to fully show the semiconvergence in relative error curves.

Further incorporating regularization on the projected problems with optimal regularization parameters, we present the same two examples solved using igenHyBR in Figure 7. Here, we observe better reconstructed images and more stable relative error curves. However, though both cases end at the same degree of inexactness with $\alpha_{50} = 10^{-6}$, igenHyBR in Figure 7a achieves better performance than in Figure 7b, with the relative error curve overlapping with its exact counterpart. This indicates that the starting degree of inexactness is decisive in reconstruction performance.



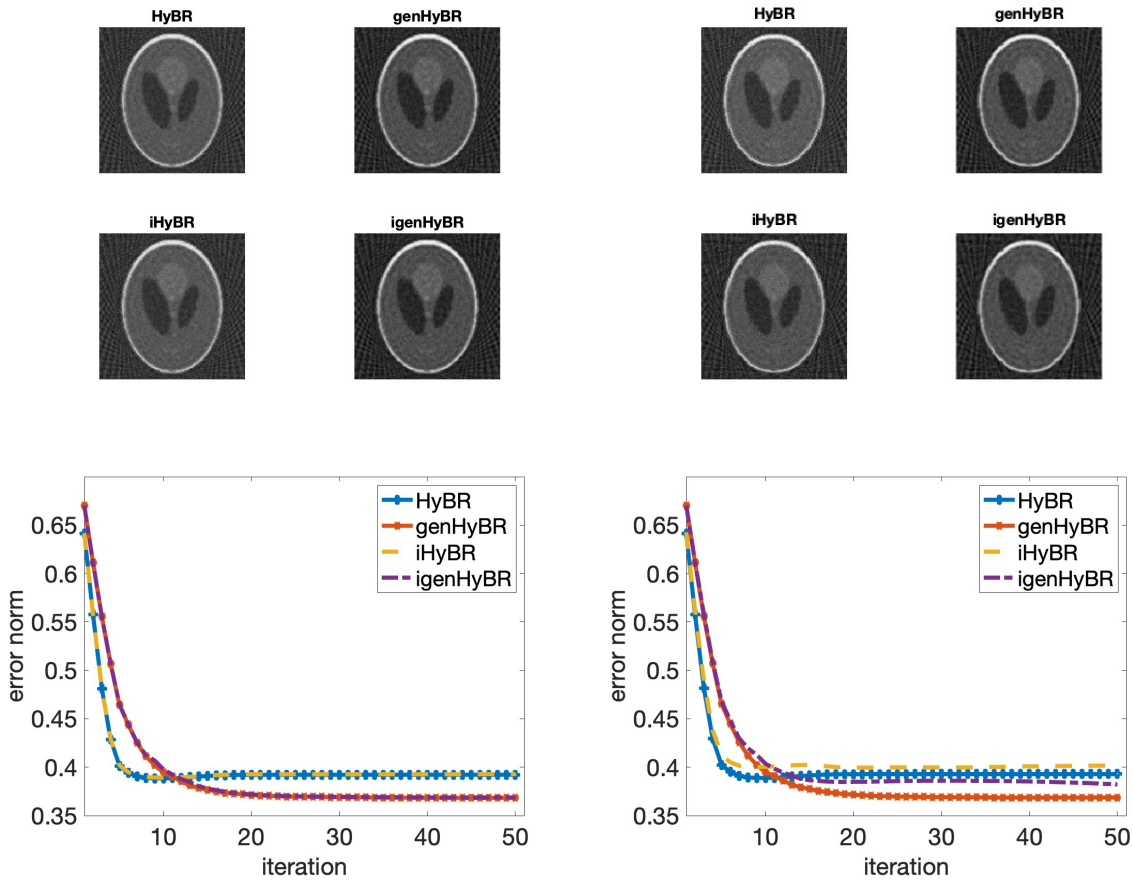
(a) Small initial inexactness

(b) Large initial inexactness

Figure 6: Comparisons of reconstruction results for sinograms acquired at inexact angles, solved without regularization. (a) starts from smaller degree of inexactness ($\alpha_1 = 10^{-1}$), (b) starts from larger degree of inexactness ($\alpha_1 = 1$).

5.5. Numerical experiment 5: Inexactness in Seismic Tomography. To further evaluate the broader applicability of our proposed method, we consider another example inverse problem that arises in seismic travel-time tomography. We use the 'PRseismic' example from [11], where the goal is to reconstruct the slowness in a given medium, which is defined as the reciprocal of the seismic velocity. With $s = 50$ sources and $p = 100$ receivers, a total of 5000 seismic travel time measurements are recorded, and the goal is to reconstruct a slowness image of size 128×128. As before, 4% noise is added to the simulated data, and we choose the prior covariance matrix $\mathbf{Q}$ to be defined as a Matérn covariance matrix with $\nu = 1.5$ and $\ell = 0.01$.

In this experiment, we consider the inexactness arising from uncertainty in the source locations. Typically, the sources are placed at predetermined locations along one edge of the



(a) Small initial inexactness

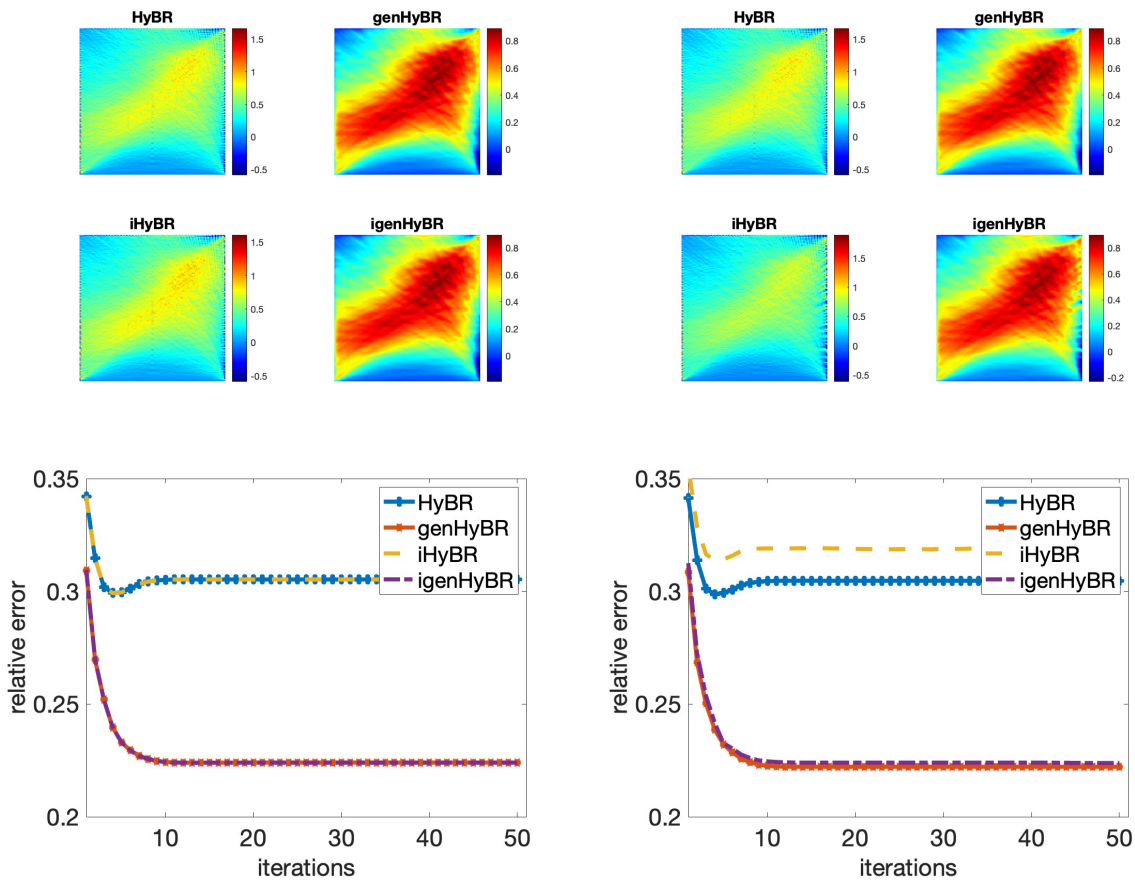
(b) Large initial inexactness

Figure 7: Comparisons of reconstruction results for sinograms acquired at inexact angles, solved through hybrid approach with optimal regularization. (a) starts from smaller degree of inexactness ($\alpha_1 = 10^{-1}$), (b) starts from larger degree of inexactness ($\alpha_1 = 1$).

domain; however, in practice, they may deviate from their intended locations due to various reasons, such as deployment errors or terrain constraints. To model this, we introduce small random perturbations to the source coordinates. That is, the x-coordinates of the sources remain fixed at 64, but the y-coordinates of the sources $\boldsymbol{\gamma}$ are perturbed as follows. Let $\boldsymbol{\gamma}_{\text{true}} \in \mathbb{R}^{50}$ denote a vector of 50 equally spaced points in $[-62.7, 62.7]$, then we simulate the inexactness at the k-th iteration through a set of perturbed y-coordinates defined as $\boldsymbol{\gamma}_k = \boldsymbol{\gamma}_{\text{true}} + \alpha_k \mathbf{e}_k$, where α_k decreases logarithmically as k increases and $\mathbf{e}_k$ is a random vector drawn from the standard normal distribution at each k. Similar to experiment setup in subsection 5.4, we consider two scenarios: the first case is presented in Figure 8a which starts with a relative small degree of inexactness $\alpha_1 = 10^{-1}$, while the second case, as presented in Figure 8b, starts from a larger degree of inexactness with $\alpha_1 = 1$. Both cases conclude

with the same $\alpha_{50} = 10^{-6}$. We provide results for the inexact hybrid approaches iHyBR and igenHyBR, and provide results for the hybrid approaches HyBR and genHyBR for comparison. All results use regularization parameters selected using the discrepancy principle.

In general, relative error curves in Figure 8 present a similar pattern as we observed above. In the small initial inexactness scenario, the error curves of inexact methods almost overlap with their exact counterparts. While in the larger inexactness scenario, the error curves of HyBR and iHyBR present larger deviations. Notably, our proposed iHyBR and igenHyBR methods produce reconstructions with a comparable level of accuracy to the HyBR and genHyBR reconstructions. Additionally, the generalized methods achieve lower relative error norms than the standard methods in the seismic tomography example, pointing to the important role of the prior covariance matrix for this example.



(a) Small initial inexactness

(b) Large initial inexactness

Figure 8: Comparisons of reconstruction results for the seismic tomography problem with inexact source locations. (a) starts from smaller degree of inexactness ($\alpha_1 = 10^{-1}$), (b) starts from larger degree of inexactness ($\alpha_1 = 1$).

6. Conclusion. In this paper, we developed an inexact generalized hybrid iterative method for efficiently computing solutions to large-scale Bayesian inverse problems with inexactness in the forward process. Unlike previous approaches which assume the exact MVPs with the forward matrix are achievable, our method adapts to the inexactness and allows effective and efficient computations. Moreover, we developed a hybrid iterative projection method that combines the igenGK projection approach with Tikhonov regularization on the projected problem. Compared to approaches that use the exact forward model, the inexact genGK methods demonstrate stability and accuracy, even for problems with large degrees of inexactness. Numerical results on tomographic image reconstruction problems validate the effectiveness of the approach and show its adaptability to real-world conditions, such as inaccurate projection angles in CT imaging and inexact source locations in seismic tomography.

Acknowledgments. The author received the Wilson Family Undergraduate STEM Research Award and would like to thank the donors. Also, this work was partially supported by the National Science Foundation program under grant DMS-2411197. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

[1] M. Arioli, *Generalized Golub–Kahan bidiagonalization and stopping criteria*, SIAM Journal on Matrix Analysis and Applications, 34 (2013), pp. 571–592, https://doi.org/10.1137/120866543.
[2] D. Calvetti, *Preconditioned iterative methods for linear discrete ill-posed problems from a Bayesian inversion perspective*, Journal of Computational and Applied Mathematics, 198 (2007), pp. 378–395.
[3] D. Calvetti and E. Somersalo, *Bayesian scientific computing*, Springer, 2023.
[4] J. Chung and S. Gazzola, *Computational methods for large-scale inverse problems: a survey on hybrid projection methods*, SIAM Review, 66 (2024), pp. 205–284.
[5] J. Chung, E. Haber, and J. Nagy, *Numerical methods for coupled super-resolution*, Inverse Problems, 22 (2006), p. 1261.
[6] J. Chung and J. G. Nagy, *An efficient iterative approach for large-scale separable nonlinear inverse problems*, SIAM Journal on Scientific Computing, 31 (2010), pp. 4654–4674.
[7] J. Chung, J. G. Nagy, D. P. O'Leary, et al., *A weighted GCV method for Lanczos hybrid regularization*, Electronic Transactions on Numerical Analysis, 28 (2008), p. 2008.
[8] J. Chung and K. Palmer, *A hybrid LSMR algorithm for large-scale Tikhonov regularization*, SIAM Journal on Scientific Computing, 37 (2015), pp. S562–S580.
[9] J. Chung and A. K. Saibaba, *Generalized hybrid iterative methods for large-scale Bayesian inverse problems*, SIAM Journal on Scientific Computing, 39 (2017), pp. S24–S46, https://doi.org/10.1137/16M1081968.
[10] T. Elfving and P. C. Hansen, *Unmatched projector/backprojector pairs: Perturbation and convergence analysis*, SIAM Journal on Scientific Computing, 40 (2018), pp. A573–A591.
[11] S. Gazzola, P. C. Hansen, and J. G. Nagy, *IR Tools: a MATLAB package of iterative regularization methods and large-scale test problems*, Numerical Algorithms, 81 (2019), pp. 773–811.
[12] S. Gazzola and M. S. Landman, *Regularization by inexact Krylov methods with applications to blind deblurring*, SIAM Journal on Matrix Analysis and Applications, 42 (2021), pp. 1528–1552, https://doi.org/10.1137/21M1402066.
[13] S. Gazzola and J. G. Nagy, *Generalized Arnoldi–Tikhonov method for sparse reconstruction*, SIAM Journal on Scientific Computing, 36 (2014), pp. B225–B247.
[14] G. Golub and W. Kahan, *Calculating the singular values and pseudo-inverse of a matrix*, Journal of the Society for Industrial and Applied Mathematics, Series B: Numerical Analysis, 2 (1965), pp. 205–224.
[15] G. Golub and V. Pereyra, *Separable nonlinear least squares: the variable projection method and its*

applications, Inverse problems, 19 (2003), p. R1.
[16] M. Hanke and P. C. Hansen, *Regularization methods for large-scale problems*, Surv. Math. Ind, 3 (1993), pp. 253–315.
[17] P. C. Hansen, *Discrete inverse problems: Insight and algorithms*, SIAM, 2010.
[18] H. Ji and K. Wang, *Robust image deblurring with an inaccurate blur kernel*, IEEE Transactions on Image processing, 21 (2011), pp. 1624–1634.
[19] M. E. Kilmer, P. C. Hansen, and M. I. Espanol, *A projection-based approach to general-form Tikhonov regularization*, SIAM Journal on Scientific Computing, 29 (2007), pp. 315–330.
[20] M. E. Kilmer and D. P. O'Leary, *Choosing regularization parameters in iterative methods for ill-posed problems*, SIAM Journal on matrix analysis and applications, 22 (2001), pp. 1204–1221.
[21] W. Nowak, S. Tenkleve, and O. A. Cirpka, *Efficient computation of linearized cross-covariance and auto-covariance matrices of interdependent quantities*, Mathematical Geology, 35 (2003), pp. 53–66.
[22] C. C. Paige and M. A. Saunders, *Algorithm 583: LSQR: Sparse linear equations and least squares problems*, ACM Transactions on Mathematical Software (TOMS), 8 (1982), pp. 195–209.
[23] C. C. Paige and M. A. Saunders, *LSQR: An algorithm for sparse linear equations and sparse least squares*, ACM Transactions on Mathematical Software (TOMS), 8 (1982), pp. 43–71.
[24] L. Reichel, F. Sgallari, and Q. Ye, *Tikhonov regularization based on generalized Krylov subspace methods*, Applied Numerical Mathematics, 62 (2012), pp. 1215–1228.
[25] R. A. Renaut, I. Hnětynková, and J. Mead, *Regularization parameter estimation for large-scale Tikhonov regularization using a priori information*, Computational statistics & data analysis, 54 (2010), pp. 3430–3445.
[26] R. A. Renaut, S. Vatankhah, and V. E. Ardestani, *Hybrid and iteratively reweighted regularization by unbiased predictive risk and weighted GCV for projected systems*, SIAM Journal on Scientific Computing, 39 (2017), pp. B221–B243.
[27] V. Simoncini and D. B. Szyld, *Theory of inexact Krylov subspace methods and applications to scientific computing*, SIAM Journal on Scientific Computing, 25 (2003), pp. 454–477.
[28] R. C. Smith, *Uncertainty quantification: theory, implementation, and applications*, SIAM, 2024.
[29] F. Uribe, J. M. Bardsley, Y. Dong, P. C. Hansen, and N. A. Riis, *A hybrid Gibbs sampler for edge-preserving tomographic reconstruction with uncertain view angles*, SIAM/ASA Journal on Uncertainty Quantification, 10 (2022), pp. 1293–1320.
[30] F. G. Waqar, S. Patel, and C. M. Simon, *A tutorial on the Bayesian statistical approach to inverse problems*, APL Machine Learning, 1 (2023), p. 041101, https://doi.org/10.1063/5.0154773, https://doi.org/10.1063/5.0154773.