



Efficient iterative methods for hyperparameter estimation in large-scale linear inverse problems

Khalil A. Hall-Hooper¹ · Arvind K. Saibaba¹  · Julianne Chung² ·
Scot M. Miller³

Received: 4 January 2024 / Accepted: 3 November 2024

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2024

Abstract

We study Bayesian methods for large-scale linear inverse problems, focusing on the challenging task of hyperparameter estimation. Typical hierarchical Bayesian formulations that follow a Markov Chain Monte Carlo approach are possible for small problems but are not computationally feasible for problems with a very large number of unknown inverse parameters. In this work, we describe an empirical Bayes (EB) method to estimate hyperparameters that maximize the marginal posterior, i.e., the probability density of the hyperparameters conditioned on the data, and then we use the estimated hyperparameters to compute the posterior of the unknown inverse parameters. For problems where the computation of the square root and inverse of prior covariance matrices are not feasible, we describe an approach based on the generalized Golub-Kahan bidiagonalization to approximate the marginal posterior and seek hyperparameters that minimize the approximate marginal posterior. Numerical results from seismic and atmospheric tomography demonstrate the accuracy, robustness, and potential benefits of the proposed approach.

Communicated by: Bangti Jin

✉ Arvind K. Saibaba
asaibab@ncsu.edu

Khalil A. Hall-Hooper
kahallho@ncsu.edu

Julianne Chung
jmchung@emory.edu

Scot M. Miller
smill191@jhu.edu

¹ Department of Mathematics, North Carolina State University, 2311 Sullivan Drive, Raleigh, NC 27695, USA

² Department of Mathematics, Emory University, 400 Dowman Drive, Atlanta, GA 30322, USA

³ Department of Environmental Health and Engineering, Johns Hopkins University, 3400 N. Charles Street, Baltimore, MD 21218, USA

Keywords Bayesian inverse problems · Hyperparameter estimation · Golub-Kahan bidiagonalization · Low-rank approximation

Mathematics Subject Classification (2010) 65F22 · 65F55 · 65M32 · 62F10 · 86A22

1 Introduction

Inverse problems arise in many important applications, where the main goal is to infer some unknown inverse parameters from given observed data. Bayesian approaches provide a robust framework for tackling such problems, where the posterior distribution of the unknown parameters combines information provided by the observations (e.g., the likelihood function) and prior knowledge about the unknown inverse parameters (e.g., the prior). However, for problems where the prior and/or the likelihood contain unspecified hyperparameters, *hierarchical* Bayesian approaches can be used such that all parameters (including the inverse parameters and the hyperparameters) can be inferred from the data in a “fully Bayesian” framework.

There are several challenges, especially for large-scale inverse problems where the number of unknowns and the size of the observed data sets are large. One of the key challenges is model specification, whereby appropriate prior distributions (including hyperpriors) and hyperparameters for each level of the hierarchy must be specified. It is important to select suitable prior distributions that capture the characteristics of the problem accurately, but finding informative and realistic priors becomes more difficult as the complexity and size of the problem increase. Another key challenge of a fully Bayesian framework is that even for linear-Gaussian inverse problems, where the forward model is linear and both the prior and error distributions are Gaussian, the posterior distribution may no longer be Gaussian, making sampling approaches significantly more challenging. These challenges are compounded by the need to perform multiple inference runs for multiple parameters, each of which requires an accurate and efficient solver for the inverse problem.

To circumvent some of the challenges of fully Bayesian approaches, empirical Bayes (EB) approaches can be used that first estimate the hyperparameters by maximizing a marginal likelihood function, and then use the estimated hyperparameters to compute the posterior of the inversion parameters. However, optimization for hyperparameters remains a challenging and expensive task (e.g., due to multiple costly objective function and gradient evaluations, each of which requires accurate inference), and much less attention in the literature has been given to estimating hyperparameters that govern prior distributions.

We focus on large-scale linear-Gaussian inverse problems, where the forward and adjoint models as well as the prior covariance matrix are only accessible via matrix-vector multiplications. This circumstance may also arise from covariance kernels that are defined on irregular grids or come from a dictionary collection. Covariance kernels from the Matérn class will be considered, although various classes of parameterized covariance kernels could be considered in this framework. For such scenarios, generalized Golub-Kahan (genGK) iterative methods have been proposed for inference [1], but the estimation of the hyperparameters is much more challenging.

Overview of Main Contributions

In this paper, we exploit Krylov subspace projections for hyperparameter estimation using empirical Bayes methods. The main contributions are as follows:

1. We propose an approximation to the marginal posterior objective function and its gradient using the genGK bidiagonalization. The genGK approach only requires matrix-vector products (matvecs) involving the forward and adjoint operators, as well as with the prior covariance matrix. However, it doesn't require applying square roots and inverses.
2. We provide a detailed analysis of the computational costs and the resulting error for the approximations. These approximations and theoretical bounds can be used for other tasks related to Bayesian inverse problems.
3. We develop methods for monitoring the accuracy of the resulting approximations and analysis for trace estimators for nonsymmetric matrices.
4. While our proposed approach is general, we show how to significantly accelerate the estimation of hyperparameters when the noise and prior variance are assumed to be unknown. This special case is important in many applications.

Other applications of our approach

Although we focus on the empirical Bayes approach, there are numerous scenarios in large-scale Bayesian inverse problems where the approximations described in this work can be used to make computational advances. For example,

1. **Fully-Bayesian approach:** Our approach provides efficient ways to approximate the objective function (obtained by the negative log-likelihood of the marginal posterior distribution) and its gradient using the genGK approaches. However, it is easy to see that the approaches can be easily extended to approximate the marginalized likelihood (see (2) for a definition). In particular, these approximations can be used to significantly accelerate Markov Chain Monte Carlo (MCMC) approaches for exploring the marginal posterior distribution and providing uncertainty estimates for the hyperparameters.
2. **Variational Bayes approach:** In this approach, we approximate the marginal posterior distribution by minimizing the distance to another class of distributions (e.g., Gaussian). As with the empirical Bayes approach, the optimization techniques also require repeated evaluations of the objective function and its gradient. Therefore, the computational techniques developed here are also applicable to that setting.
3. **Information-theoretic approaches:** An alternative approach to estimating the hyperparameters is to optimize the information gain from the prior distribution to the posterior. One such metric is the expected Kullback-Liebler divergence, also known as the D-optimal design criterion, which has a similar form as our objective function (in fact, it only involves the log-determinant term). Therefore, the approximation techniques developed here could also be applied to estimate hyperparameters using information theoretic criteria.

1.1 Related work

Hyperparameter estimation in hierarchical Bayesian problems is an important problem that has received some attention. In the geostatistical community, an approach

for estimating hyperparameters is the maximum likelihood and restricted maximum likelihood approaches [2–4]. However, these approaches are not scalable to large-scale problems and are different from ours, since we use the MAP estimate of the marginalized posterior.

In contrast to optimization-based techniques, sampling methods based on MCMC have been used to estimate the uncertainty in the hyperparameters (see, e.g., [5]). However, this approach is considerably more expensive since it requires computing the likelihood thousands of times to get reliable samples from the posterior distribution. Another approach to estimating hyperparameters is using the semivariogram technique [6], but this is only applicable to estimating the prior hyperparameters.

Our work is close in spirit to [7] which also uses a low-rank approximation to accelerate the computation of the objective function based on a marginal likelihood; to find the optimal parameters, they evaluated it over a finite grid of hyperparameters. Our approach instead uses the marginal posterior to construct the objective function and uses the genGK iterative method to approximate it. Furthermore, we show how to approximate the derivatives using the genGK approach and use it in a continuous optimization approach.

In previous work, we developed a solver called genHyBR, also based on genGK, for Bayesian inverse problems [1] but it was only capable of estimating a limited set of hyperparameters (associated with the prior). The authors in [8] also use the genGK approach in Gaussian processes, similar to ours, to estimate hyperparameters but use the profile likelihood which is not a Bayesian approach like ours.

1.2 Outline of the paper

The paper is organized as follows. In Section 2, we provide a brief overview of hyperparameter estimation, from a hierarchical Bayes formulation to an empirical Bayes approach. We describe the generalized Golub-Kahan bidiagonalization process, which will be used in Section 3. That is, we use elements from the genGK bidiagonalization to approximate the EB objective function and gradient for optimization and provide an error analysis for approximations that can be used to monitor the accuracy of approximations. We demonstrate the performance of our approach on model inverse problems in heat conduction, seismic and atmospheric tomography in Section 4, and conclusions and future work are provided in Section 5. Theoretical derivations have been relegated to the Appendix.

2 Background and problem Setup

2.1 Notation

Let m and n be strictly positive integers, and denote the nonnegative reals by \mathbb{R}_+ . In the following, \mathbb{R}^n is the standard n -dimensional Euclidean space with scalar product $(\mathbf{x}, \mathbf{y}) \mapsto \langle \mathbf{x}, \mathbf{y} \rangle_2 = \mathbf{y}^\top \mathbf{x}$, and $\mathcal{M}_{m,n}(\mathbb{R})$ is the normed space of all real-valued m -by- n matrices with scalar product $(\mathbf{A}, \mathbf{B}) \mapsto \langle \mathbf{A}, \mathbf{B} \rangle_F = \text{tr}(\mathbf{B}^\top \mathbf{A})$, where \top denotes the matrix transpose. We write $\mathcal{M}_{m,m}(\mathbb{R}) \equiv \mathcal{M}_m(\mathbb{R})$. Let $\mathbf{A} \in \mathcal{M}_n(\mathbb{R})$, and let \mathbf{x} and

\mathbf{y} be in \mathbb{R}^n . If \mathbf{A} is symmetric positive definite (SPD), we write $\langle \mathbf{x}, \mathbf{y} \rangle_{\mathbf{A}} \equiv \langle \mathbf{Ax}, \mathbf{y} \rangle_2$ and $\|\mathbf{x}\|_{\mathbf{A}} \equiv \sqrt{\langle \mathbf{Ax}, \mathbf{x} \rangle_2}$, with $\|\mathbf{x}\|_2 \equiv \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle_2}$. We also write $\|\mathbf{A}\|_F \equiv \sqrt{\langle \mathbf{A}, \mathbf{A} \rangle_F}$ to denote the Frobenius norm and $\|\mathbf{A}\|_2$ denotes the spectral or 2-norm.

Given any realization, say δ , of a random variable, we write X_{δ} to denote said variable. We also use the notation $\pi[\cdot]$ to denote an arbitrary probability density.

Let \mathcal{X} be a real vector space, $f : \mathcal{X} \rightarrow \mathbb{R}$, and let $x \in \mathcal{X}$. Denote by $\text{Argmin } f \equiv \{x \in \mathcal{X} \mid f(x) = \min_{z \in \mathcal{X}} f(z)\}$. If $\text{Argmin } f$ is single-valued, then we write that unique element as $\text{argmin}_{x \in \mathcal{X}} f$.

2.2 Hierarchical Bayes formulation

Let m , n , and K be strictly positive integers with $m \leq n$. We consider a linear inverse problem of recovering an unknown vector $\mathbf{s} \in \mathbb{R}^n$ of signals from a noisy vector $\mathbf{d} \in \mathbb{R}^m$ of data observations. The signals and observations are characterized by the model

$$\mathbf{d} = \mathbf{As} + \boldsymbol{\eta}, \quad (1)$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$ represents the forward map and $\boldsymbol{\eta} \in \mathbb{R}^m$ is a vector of realizations of a random variable $X_{\boldsymbol{\eta}} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}(\cdot))$. In this setting, we assume that $\mathbf{R} : (0, +\infty)^K \rightarrow \mathbb{R}^{m \times m}$, where $\mathbf{R}(\cdot)$ is SPD, is uncorrelated; this ensures that $\mathbf{R}(\cdot)$ is a diagonal matrix, so it is easy to invert and compute the square root.

Let $\boldsymbol{\theta} = (\theta_i)_{1 \leq i \leq K} \in (0, +\infty)^K$ be a vector of unknown hyperparameters, with prior density $\pi[\boldsymbol{\theta}]$. Following a hierarchical Bayesian approach, we model $\mathbf{s} | \boldsymbol{\theta}$ as a realization of a Gaussian random variable $X_{\mathbf{s} | \boldsymbol{\theta}} \sim \mathcal{N}(\boldsymbol{\mu}(\boldsymbol{\theta}), \mathbf{Q}(\boldsymbol{\theta}))$, where $\boldsymbol{\mu} : (0, +\infty)^K \rightarrow \mathbb{R}^n$, $\mathbf{Q} : (0, +\infty)^K \rightarrow \mathbb{R}^{n \times n}$, and $\mathbf{Q}(\boldsymbol{\theta})$ is SPD, but is computationally infeasible to compute explicitly; therefore, we do not assume that we have access to its square root or its inverse. We do, however, assume that the matvecs with $\mathbf{Q}(\boldsymbol{\theta})$ can be performed easily in $\mathcal{O}(n \log n)$ time by using the fast Fourier Transform (FFT), if the solution is represented on a uniform equispaced grid [9], or by using hierarchical matrices [10]; furthermore, $\mathcal{O}(n)$ time can be achieved in certain circumstances with the aid of \mathcal{H}^2 -matrices or the fast multipole method [11].

Using Bayes' theorem, the posterior density, $\pi[\mathbf{s}, \boldsymbol{\theta} | \mathbf{d}]$, is characterized by

$$\pi[\mathbf{s}, \boldsymbol{\theta} | \mathbf{d}] = \frac{\pi[\mathbf{d} | \mathbf{s}, \boldsymbol{\theta}] \pi[\mathbf{s} | \boldsymbol{\theta}] \pi[\boldsymbol{\theta}]}{\pi[\mathbf{d}]}.$$

In our setting, the posterior is explicitly

$$\pi[\mathbf{s}, \boldsymbol{\theta} | \mathbf{d}] \propto \frac{\pi[\boldsymbol{\theta}] \exp\left(-\frac{1}{2} \|\mathbf{d} - \mathbf{As}\|_{\mathbf{R}^{-1}(\boldsymbol{\theta})}^2 - \frac{1}{2} \|\mathbf{s} - \boldsymbol{\mu}(\boldsymbol{\theta})\|_{\mathbf{Q}^{-1}(\boldsymbol{\theta})}^2\right)}{\det(\mathbf{R}(\boldsymbol{\theta}))^{1/2} \det(\mathbf{Q}(\boldsymbol{\theta}))^{1/2}},$$

where \propto denotes the proportionality relation. The marginal posterior density is obtained by integrating out the unknowns \mathbf{s} and can be represented as

$$\pi[\boldsymbol{\theta} | \mathbf{d}] \propto \pi[\boldsymbol{\theta}] \det(\mathbf{Z}(\boldsymbol{\theta}))^{-1/2} \exp\left(-\frac{1}{2} \|\mathbf{A}\boldsymbol{\mu}(\boldsymbol{\theta}) - \mathbf{d}\|_{\mathbf{Z}^{-1}(\boldsymbol{\theta})}^2\right), \quad (2)$$

where $\mathbf{Z} : (0, +\infty)^K \rightarrow \mathbb{R}^{m \times m}$ has the representation

$$\mathbf{Z}(\boldsymbol{\theta}) = \mathbf{A}\mathbf{Q}(\boldsymbol{\theta})\mathbf{A}^\top + \mathbf{R}(\boldsymbol{\theta}), \quad (3)$$

see, e.g., [12, 13].

For the case where $\boldsymbol{\theta}$ is a *fixed*, known value, or in the non-hierarchical setting, the maximum a posteriori estimate \mathbf{s}_{post} , which is obtained by minimizing the negative logarithm of the posterior distribution, i.e., $\pi_{\text{post}}[\mathbf{s} | \mathbf{d}] \propto \pi[\mathbf{s} | \mathbf{d}] \pi[\mathbf{s}]$ (see, e.g., [14]) and has the explicit expression

$$\begin{aligned} \mathbf{s}_{\text{post}} &= \operatorname{argmin}_{\mathbf{s} \in \mathbb{R}^n} (-\log \pi_{\text{post}}[\mathbf{s} | \mathbf{d}]) \\ &= \operatorname{argmin}_{\mathbf{s} \in \mathbb{R}^n} \frac{1}{2} \|\mathbf{d} - \mathbf{A}\mathbf{s}\|_{\mathbf{R}^{-1}}^2 + \frac{1}{2} \|\mathbf{s} - \boldsymbol{\mu}\|_{\mathbf{Q}^{-1}}^2, \end{aligned}$$

where we have suppressed the arguments containing $\boldsymbol{\theta}$ to indicate that these quantities are fixed. The MAP estimate \mathbf{s}_{post} can equivalently be expressed in closed form as

$$\mathbf{s}_{\text{post}} = \boldsymbol{\Gamma}_{\text{post}}(\mathbf{A}^\top \mathbf{R}^{-1} \mathbf{d} + \mathbf{Q}^{-1} \boldsymbol{\mu}), \quad (4)$$

where $\boldsymbol{\Gamma}_{\text{post}} = (\mathbf{A}^\top \mathbf{R}^{-1} \mathbf{A} + \mathbf{Q}^{-1})^{-1}$. For large inverse problems, solving (4) is not computationally feasible in practice, and several studies provide alternative, iterative approaches.

2.3 The empirical Bayes method for hyperparameter estimation

The empirical Bayes framework (also known as evidence approximation in the machine learning literature) allows one to estimate the values of the hyperparameters via the marginal posterior. In doing so, the choice of the hyperparameters is informed by the model, the data, and any assumptions about the likelihood and prior formulations. The main idea behind the EB method [15] is to estimate the hyperparameters $\boldsymbol{\theta}_{\text{EB}}$ and fix the hyperparameters in the posterior distribution. That is, we set $\pi[\mathbf{s} | \mathbf{d}] \approx \pi[\mathbf{s}, \boldsymbol{\theta}_{\text{EB}} | \mathbf{d}]$. The EB approach has some known drawbacks: first, it uses the data twice, once to estimate the hyperparameters and second to determine the unknowns \mathbf{s} , second, it ignores the uncertainty in the hyperparameters and, therefore, underestimates the overall uncertainty. Nevertheless, it remains a popular approach in statistics, especially for computationally challenging problems, and we adopt this approach in the present paper.

Specifically, we choose the hyperparameters $\boldsymbol{\theta}_{\text{EB}}$ obtained from the maximum a posteriori (MAP) estimate of the marginal posterior distribution (2); alternatively, it can be obtained by minimizing the objective function $\mathcal{F}(\boldsymbol{\theta})$, defined by

$$\mathcal{F}(\boldsymbol{\theta}) = -\log \pi[\boldsymbol{\theta}] + \frac{1}{2} \operatorname{logdet}(\mathbf{Z}(\boldsymbol{\theta})) + \frac{1}{2} \|\mathbf{A}\boldsymbol{\mu}(\boldsymbol{\theta}) - \mathbf{d}\|_{\mathbf{Z}^{-1}(\boldsymbol{\theta})}^2. \quad (5)$$

Once the optimal hyperparameters $\boldsymbol{\theta}_{\text{EB}}$ have been estimated, we can obtain the MAP estimate for \mathbf{s} by (4).

Anticipating that we will use a gradient-based approach for optimizing (5), we provide an analytical expression for the gradient $\nabla \mathcal{F} = (\frac{\partial \mathcal{F}}{\partial \theta_i})_{1 \leq i \leq K}$,

$$\begin{aligned} (\nabla \mathcal{F}(\boldsymbol{\theta}))_i = & -\frac{1}{\pi[\boldsymbol{\theta}]} \frac{\partial \pi[\boldsymbol{\theta}]}{\partial \theta_i} + \frac{1}{2} \left\langle \mathbf{Z}^{-1}(\boldsymbol{\theta}), \frac{\partial \mathbf{Z}(\boldsymbol{\theta})}{\partial \theta_i} \right\rangle_F \\ & - \frac{1}{2} \left[\mathbf{Z}^{-1}(\boldsymbol{\theta})(\mathbf{A}\boldsymbol{\mu}(\boldsymbol{\theta}) - \mathbf{d}) \right]^\top \left[\frac{\partial \mathbf{Z}(\boldsymbol{\theta})}{\partial \theta_i} \mathbf{Z}^{-1}(\boldsymbol{\theta})(\mathbf{A}\boldsymbol{\mu}(\boldsymbol{\theta}) - \mathbf{d}) - \mathbf{A} \frac{\partial \boldsymbol{\mu}(\boldsymbol{\theta})}{\partial \theta_i} \right]. \end{aligned} \quad (6)$$

Notice that there are various challenges for optimization. First, the objective function (\mathcal{F}) is nonconvex, with potentially many local minima, so this causes problems for the optimization algorithms which can be sensitive to the choice of the initial guess. Second, computing the objective function and the gradient ($\nabla \mathcal{F}$) for each new candidate set of hyperparameters $\boldsymbol{\theta}$ requires recalculating the log determinant and the inverse of an $m \times m$ matrix; even forming this matrix explicitly is expensive and should be avoided. This can become computationally prohibitive for problems with large observational datasets. We focus our attention in this paper on the second challenge, and in Section 3 propose efficient methods to compute the objective function and the gradient. Our approach is based on low-rank approximations of the matrix \mathbf{A} , which we achieve using the generalized Golub-Kahan (genGK) bidiagonalization. This approach is discussed next. The low-rank approximations make it easier to evaluate the objective function and the gradient.

2.4 Generalized Golub-Kahan bidiagonalization

The generalized Golub-Kahan (genGK) bidiagonalization was used for efficiently computing Tikhonov regularized solutions (4) in [1] and for inverse UQ in [16]. However, in this work, we extend the use of the genGK bidiagonalization process for efficient hyperparameter estimation following an EB approach. For completeness, we provide an overview of the genGK process.

Throughout this section, suppose we have some fixed hyperparameters $\bar{\boldsymbol{\theta}} \in \mathbb{R}_+^K$. Let $\mathbf{R} \equiv \mathbf{R}(\bar{\boldsymbol{\theta}})$, $\mathbf{Q} \equiv \mathbf{Q}(\bar{\boldsymbol{\theta}})$, and $\boldsymbol{\mu} = \boldsymbol{\mu}(\bar{\boldsymbol{\theta}})$. Given the matrices \mathbf{A} , \mathbf{R} , \mathbf{Q} and the vector \mathbf{d} (as defined in (1)), with the initialization

$$\beta_1 \mathbf{u}_1 = \mathbf{d} - \mathbf{A}\boldsymbol{\mu} \quad \alpha_1 \mathbf{v}_1 = \mathbf{A}^\top \mathbf{R}^{-1} \mathbf{u}_1$$

the j th iteration of the genGK (lower) bidiagonalization process generates vectors \mathbf{u}_{j+1} and \mathbf{v}_{j+1} such that

$$\begin{aligned} \beta_{j+1} \mathbf{u}_{j+1} &= \mathbf{A}\mathbf{Q}\mathbf{v}_j - \alpha_j \mathbf{u}_j; \\ \alpha_{j+1} \mathbf{v}_{j+1} &= \mathbf{A}^\top \mathbf{R}^{-1} \mathbf{u}_{j+1} - \beta_{j+1} \mathbf{v}_j, \end{aligned}$$

where, for some natural number i , the scalars $(\alpha_i, \beta_i) \in \mathbb{R}_+^2$ are chosen such that $\|\mathbf{u}_i\|_{\mathbf{R}^{-1}} = \|\mathbf{v}_i\|_{\mathbf{Q}} = 1$. At the end of k iterations, we have a lower bidiagonal matrix

$$\mathbf{B}_k = \begin{bmatrix} \alpha_1 & & & \\ \beta_2 & \alpha_2 & & \\ & \ddots & \ddots & \\ & & \ddots & \alpha_k \\ & & & \beta_{k+1} \end{bmatrix} \in \mathbb{R}^{(k+1) \times k},$$

and two matrices $\mathbf{U}_{k+1} = [\mathbf{u}_1 \mid \cdots \mid \mathbf{u}_{k+1}] \in \mathbb{R}^{m \times (k+1)}$ and $\mathbf{V}_k = [\mathbf{v}_1 \mid \mathbf{v}_2 \mid \cdots \mid \mathbf{v}_k] \in \mathbb{R}^{n \times k}$ that satisfy the orthogonality relations

$$\mathbf{U}_{k+1}^\top \mathbf{R}^{-1} \mathbf{U}_{k+1} = \mathbf{I}_{k+1} \quad \text{and} \quad \mathbf{V}_k^\top \mathbf{Q} \mathbf{V}_k = \mathbf{I}_k. \quad (7)$$

Furthermore, these matrices satisfy the genGK relations

$$\begin{cases} \mathbf{U}_{k+1} \beta_1 \mathbf{e}_1 &= \mathbf{d} - \mathbf{A} \boldsymbol{\mu} \\ \mathbf{A} \mathbf{Q} \mathbf{V}_k &= \mathbf{U}_{k+1} \mathbf{B}_k \\ \mathbf{A}^\top \mathbf{R}^{-1} \mathbf{U}_{k+1} &= \mathbf{V}_k \mathbf{B}_k^\top + \alpha_{k+1} \mathbf{v}_{k+1} \mathbf{e}_{k+1}^\top. \end{cases} \quad (8)$$

Here, \mathbf{e}_j represents the j th column of the identity matrix of appropriate size.

In floating point arithmetic, these relations are typically accurate to machine precision. However, the matrices \mathbf{U}_k and \mathbf{V}_k tend to lose orthogonality with respect to the appropriate inner products; therefore, in practice, we use complete reorthogonalization to mitigate this loss in accuracy. The above process is summarized in Algorithm 1.

Algorithm 1 genGK bidiagonalization. Call as $[\mathbf{U}_{k+1}, \mathbf{V}_k, \mathbf{B}_k] = \text{genGK}(\mathbf{A}, \mathbf{R}, \mathbf{Q}, \boldsymbol{\mu}, \mathbf{d}, k)$.

Require: Matrices \mathbf{A} , \mathbf{R} , and \mathbf{Q} ; vectors $\boldsymbol{\mu}$ and \mathbf{d} .

- 1: $\beta_1 = \|\mathbf{d} - \mathbf{A} \boldsymbol{\mu}\|_{\mathbf{R}^{-1}}$;
- 2: $\mathbf{u}_1 = \frac{\mathbf{d} - \mathbf{A} \boldsymbol{\mu}}{\beta_1}$;
- 3: $\alpha_1 = \|\mathbf{A}^\top \mathbf{R}^{-1} \mathbf{u}_1\|_{\mathbf{Q}}$;
- 4: $\mathbf{v}_1 = \frac{\mathbf{A}^\top \mathbf{R}^{-1} \mathbf{u}_1}{\alpha_1}$;
- 5: **for** $j = 1, 2, \dots, k$ **do**
- 6: $\beta_{j+1} = \|\mathbf{A} \mathbf{Q} \mathbf{v}_j - \alpha_j \mathbf{u}_j\|_{\mathbf{R}^{-1}}$;
- 7: $\mathbf{u}_{j+1} = \frac{\mathbf{A} \mathbf{Q} \mathbf{v}_j - \alpha_j \mathbf{u}_j}{\beta_{j+1}}$;
- 8: $\alpha_{j+1} = \|\mathbf{A}^\top \mathbf{R}^{-1} \mathbf{u}_{j+1} - \beta_{j+1} \mathbf{v}_j\|_{\mathbf{Q}}$;
- 9: $\mathbf{v}_{j+1} = \frac{\mathbf{A}^\top \mathbf{R}^{-1} \mathbf{u}_{j+1} - \beta_{j+1} \mathbf{v}_j}{\alpha_{j+1}}$;
- 10: **end for**
- 11: **return** Matrices \mathbf{U}_{k+1} , \mathbf{V}_k , \mathbf{B}_k .

Low-rank approximation

We can reinterpret the genGK relations through the lens of projectors which leads us to a low-rank approximation of \mathbf{A} . The low-rank approximation of \mathbf{A} will be critical in developing efficient approaches in Section 3. Define the following projectors:

$$\mathbf{P}_{\mathbf{V}_k} = \mathbf{V}_k \mathbf{V}_k^\top \mathbf{Q}; \quad \mathbf{P}_{\mathbf{U}_{k+1}} = \mathbf{U}_{k+1} \mathbf{U}_{k+1}^\top \mathbf{R}^{-1}. \quad (9)$$

Then orthogonality relations in (7) imply that $\mathbf{P}_{\mathbf{V}_k}$ and $\mathbf{P}_{\mathbf{U}_{k+1}}$ are oblique projectors. Using this insight, we can rewrite (8) as follows:

$$\begin{cases} \mathbf{U}_{k+1} \beta_1 \mathbf{e}_1 &= \mathbf{P}_{\mathbf{U}_{k+1}} (\mathbf{d} - \mathbf{A} \boldsymbol{\mu}) \\ \mathbf{A} \mathbf{P}_{\mathbf{V}_k}^\top &= \mathbf{U}_{k+1} \mathbf{B}_k \mathbf{V}_k^\top \\ \mathbf{A}^\top \mathbf{P}_{\mathbf{U}_{k+1}}^\top &= \mathbf{V}_{k+1} \begin{bmatrix} \mathbf{B}_k^\top \\ \alpha_{k+1} \mathbf{e}_{k+1}^\top \end{bmatrix} \mathbf{U}_{k+1}^\top. \end{cases} \quad (10)$$

This reinterpretation of the genGK relations in (8) using projectors suggests a natural low-rank approximation for \mathbf{A} :

$$\mathbf{A} \approx \tilde{\mathbf{A}}_k = \mathbf{A} \mathbf{P}_{\mathbf{V}_k}^\top = \mathbf{U}_{k+1} \mathbf{B}_k \mathbf{V}_k^\top. \quad (11)$$

In the next section, we will use this low-rank approximation to approximate the objective function (5) and the gradient (6) for EB hyperparameter estimation. Moreover, the projection-based viewpoint of the low-rank approximation will be useful for error analysis.

3 Efficient hyperparameter estimation

The main goal of this section is to solve the optimization problem involving (5) by exploiting the genGK bidiagonalization. Consider minimizing the marginal posterior,

$$\min_{\boldsymbol{\theta} \in \mathbb{R}_+^K} \mathcal{F}(\boldsymbol{\theta}) \quad (12)$$

where $\mathcal{F}(\boldsymbol{\theta})$ is defined in (5). First, we use the genGK bidiagonalization to approximate the objective function and gradient, so that evaluations during optimization can be done efficiently. Then, we provide an error analysis to quantify the errors between \mathcal{F} and its approximations, which will be used for monitoring the convergence of the genGK process.

3.1 Approximations using genGK

As before, we assume that we have a fixed $\bar{\boldsymbol{\theta}}$, and for ease of exposition, we suppress the dependence on $\bar{\boldsymbol{\theta}}$. For any differentiable mapping $T : \mathbb{R}_+^K \rightarrow \mathbb{R}^{m \times n}$, we write for

brevity

$$\partial_{\theta_i} T = \frac{\partial T(\boldsymbol{\theta})}{\partial \theta_i} \bigg|_{\boldsymbol{\theta}=\bar{\boldsymbol{\theta}}} \quad 1 \leq i \leq K.$$

The objective function $\bar{\mathcal{F}} \equiv \mathcal{F}(\bar{\boldsymbol{\theta}})$ takes the form

$$\bar{\mathcal{F}} \equiv -\log \pi + \frac{1}{2} \log \det(\mathbf{Z}) + \frac{1}{2} \|\mathbf{A}\boldsymbol{\mu} - \mathbf{d}\|_{\mathbf{Z}^{-1}}^2, \quad (13)$$

and the gradient $\nabla \mathcal{F} = (\partial_{\theta_i} \mathcal{F})_{1 \leq i \leq K} \in \mathbb{R}^K$ takes the form (for $1 \leq i \leq k$)

$$\begin{aligned} \partial_{\theta_i} \mathcal{F} = & -\frac{1}{\pi[\bar{\boldsymbol{\theta}}]} \frac{\partial \pi[\bar{\boldsymbol{\theta}}]}{\partial \theta_i} + \frac{1}{2} \left\langle \mathbf{Z}^{-1}, \partial_{\theta_i} \mathbf{Z}(\bar{\boldsymbol{\theta}}) \right\rangle_F \\ & - \frac{1}{2} \left\langle (\partial_{\theta_i} \mathbf{Z}) \mathbf{Z}^{-1} (\mathbf{A}\boldsymbol{\mu} - \mathbf{d}) - \mathbf{A} \partial_{\theta_i} \boldsymbol{\mu}(\bar{\boldsymbol{\theta}}), \mathbf{Z}^{-1} (\mathbf{A}\boldsymbol{\mu} - \mathbf{d}) \right\rangle_2. \end{aligned}$$

The matrix \mathbf{Z} was defined in (3).

Approximation to the objective function

We can approximate $\bar{\mathcal{F}}$ by substituting our low-rank approximation $\tilde{\mathbf{A}}_k$ for \mathbf{A} in (11) (with the exception of one term). Along with the use of the Sherman–Morrison–Woodbury formula, yields the following formulas for \mathbf{Z} and \mathbf{Z}^{-1} :

$$\mathbf{Z} \approx \tilde{\mathbf{Z}}_k = \mathbf{U}_{k+1} \mathbf{B}_k (\mathbf{U}_{k+1} \mathbf{B}_k)^\top + \mathbf{R} \quad (14)$$

$$\mathbf{Z}^{-1} \approx \tilde{\mathbf{Z}}_k^{-1} = \mathbf{R}^{-1} - \mathbf{R}^{-1} \mathbf{U}_{k+1} \mathbf{B}_k (\mathbf{I}_k + \mathbf{B}_k^\top \mathbf{B}_k)^{-1} (\mathbf{U}_{k+1} \mathbf{B}_k)^\top \mathbf{R}^{-1}. \quad (15)$$

Set $\boldsymbol{\Theta}_k = \mathbf{I}_{k+1} + \mathbf{B}_k \mathbf{B}_k^\top$, which we note is positive definite. Therefore, using Sylvester's determinant identity, we approximate $\log \det(\mathbf{Z})$ as

$$\log \det(\mathbf{Z}) \approx \log \det(\tilde{\mathbf{Z}}_k) = \log \det(\mathbf{R}) + \log \det(\boldsymbol{\Theta}_k).$$

Additionally, using (8), we can approximate

$$\|\mathbf{A}\boldsymbol{\mu} - \mathbf{d}\|_{\mathbf{Z}^{-1}}^2 = \|\mathbf{U}_{k+1} \beta_1 \mathbf{e}_1\|_{\mathbf{Z}^{-1}}^2 \approx \|\mathbf{U}_{k+1} \beta_1 \mathbf{e}_1\|_{\tilde{\mathbf{Z}}_k^{-1}}^2 = \|\beta_1 \mathbf{e}_1\|_{\boldsymbol{\Theta}_k^{-1}}^2.$$

Putting everything together, our approximation of $\bar{\mathcal{F}}$, denoted by $\tilde{\mathcal{F}}_k$, is

$$\tilde{\mathcal{F}}_k \equiv -\log \pi + \frac{1}{2} \log \det(\mathbf{R}) + \frac{1}{2} \log \det(\boldsymbol{\Theta}_k) + \frac{1}{2} \|\beta_1 \mathbf{e}_1\|_{\boldsymbol{\Theta}_k^{-1}}^2. \quad (16)$$

We observed that forming $\boldsymbol{\Theta}_k$ explicitly caused numerical issues while evaluating the log determinant term. Therefore, we instead computed this term by computing the singular values of \mathbf{B}_k , denoted $\{\sigma_j(\mathbf{B}_k)\}_{j=1}^k$, and then computing $\log \det(\boldsymbol{\Theta}_k) = \sum_{j=1}^k \log(1 + \sigma_j(\mathbf{B}_k)^2)$.

Approximation to gradient

We can, similarly, approximate our expression for the gradient when it is evaluated at the point $\bar{\theta}$. By utilizing (11), we have

$$\partial_{\theta_i} \mathbf{Z} \approx \widetilde{\partial_{\theta_i} \mathbf{Z}}_k \equiv \tilde{\mathbf{A}}_k \partial_{\theta_i} \mathbf{Q} \tilde{\mathbf{A}}_k^\top + \partial_{\theta_i} \mathbf{R}, \quad 1 \leq i \leq K. \quad (17)$$

In view of (15), our approximation to $\partial_{\theta_i} \mathcal{F}$, denoted by $\widetilde{\partial_{\theta_i} \mathcal{F}}_k$, is

$$\begin{aligned} \widetilde{\partial_{\theta_i} \mathcal{F}}_k = & -\frac{1}{\pi[\bar{\theta}]} \frac{\partial \pi[\bar{\theta}]}{\partial \theta_i} + \frac{1}{2} \left\langle \tilde{\mathbf{Z}}_k^{-1}, \widetilde{\partial_{\theta_i} \mathbf{Z}}_k \right\rangle_F \\ & - \frac{1}{2} \left\langle \left(\widetilde{\partial_{\theta_i} \mathbf{Z}}_k \right) \tilde{\mathbf{Z}}_k^{-1} (\mathbf{A}\boldsymbol{\mu} - \mathbf{d}) - \mathbf{A} \partial_{\theta_i} \boldsymbol{\mu}, \tilde{\mathbf{Z}}_k^{-1} (\mathbf{A}\boldsymbol{\mu} - \mathbf{d}) \right\rangle_F, \quad 1 \leq i \leq K. \end{aligned} \quad (18)$$

The approximation of the gradient $\nabla \mathcal{F}$ is denoted by $\widetilde{\nabla \mathcal{F}}_k$. In the form written, it is not clear how the approximations lead to improved computational benefits. However, we can derive an equivalent but alternative expression that is more computationally efficient to evaluate and that is what we implement in practice. To this end, define the matrices

$$\Psi_i^Q = \mathbf{V}_k^\top \partial_{\theta_i} \mathbf{Q} \mathbf{V}_k, \quad \Psi_i^R = \mathbf{U}_{k+1}^\top \mathbf{R}^{-1} (\partial_{\theta_i} \mathbf{R}) \mathbf{R}^{-1} \mathbf{U}_{k+1} \quad 1 \leq i \leq K, \quad (19)$$

and the matrix $\mathbf{T}_k = \mathbf{B}_k^\top \mathbf{B}_k$. With these definitions, we can show (see Appendix A)

$$\left\langle \tilde{\mathbf{Z}}_k^{-1}, \widetilde{\partial_{\theta_i} \mathbf{Z}}_k \right\rangle_F = \left\langle \Psi_i^Q, \mathbf{T}_k (\mathbf{I} + \mathbf{T}_k)^{-1} \right\rangle_F + \left\langle \partial_{\theta_i} \mathbf{R}, \mathbf{R}^{-1} \right\rangle_F - \left\langle \mathbf{B}_k^\top \Psi_i^R \mathbf{B}_k, (\mathbf{I} + \mathbf{T}_k)^{-1} \right\rangle_F. \quad (20)$$

The details of computing the estimates of the objective function and the gradient are given in Algorithm 2. In Section 3.2, we will derive computational costs and provide an error analysis.

Algorithm 2 Computing approximations to the objective function and gradient based on genGK

Require: Parameter θ , matrix \mathbf{A} , parameter k , and data \mathbf{d}

- 1: Compute matrices \mathbf{R} , \mathbf{Q} and vector $\boldsymbol{\mu}$ corresponding to the parameter θ
- 2: $[\mathbf{U}_{k+1}, \mathbf{V}_{k+1}, \mathbf{B}_k] = \text{genGK}(\mathbf{A}, \mathbf{R}, \mathbf{Q}, \boldsymbol{\mu}, \mathbf{d}, k)$;
- 3: {Step 1: Estimating the objective function}
- 4: Compute $\boldsymbol{\Theta}_k = \mathbf{I}_{k+1} + \mathbf{B}_k \mathbf{B}_k^\top$ and $\mathbf{T}_k = \mathbf{B}_k^\top \mathbf{B}_k$
- 5: Compute $\tilde{\mathcal{F}}_k$ using (16).
- 6: {Step 2: Estimating the gradient}
- 7: Compute $\mathbf{r}_k = \tilde{\mathbf{Z}}_k^{-1} (\mathbf{A}\boldsymbol{\mu} - \mathbf{d})$ using (15)
- 8: **for** $i = 1, \dots, K$ **do**
- 9: Compute Ψ_i^Q and Ψ_i^R as in (19)
- 10: Compute $\left\langle \tilde{\mathbf{Z}}_k^{-1}, \widetilde{\partial_{\theta_i} \mathbf{Z}}_k \right\rangle_F$ using (20)
- 11: Compute $\widetilde{\partial_{\theta_i} \mathbf{Z}}_k \mathbf{r}_k = \mathbf{U}_{k+1} \mathbf{B}_k \Psi_i^Q (\mathbf{U}_{k+1} \mathbf{B}_k)^\top \mathbf{r}_k + \partial_{\theta_i} \mathbf{R} \mathbf{r}_k$ by exploiting the low-rank structure
- 12: **end for**
- 13: **return** Approximations $\tilde{\mathcal{F}}_k$ in (16) and $(\widetilde{\partial_{\theta_i} \mathcal{F}}_{i=1}^K)$ in (18).

Table 1 Computational costs associated with Algorithm 1, $\tilde{\mathcal{F}}_k$, and $\partial\tilde{\mathcal{F}}_k$

	genGK	$\tilde{\mathcal{F}}_k$	$\partial\tilde{\mathcal{F}}_k$
Involving \mathbf{A}	$2(k+1)\mathcal{T}_{\mathbf{A}}$	0	0
Involving \mathbf{Q} (and derivatives)	$(2k+1)\mathcal{T}_{\mathbf{Q}}$	0	$kK\mathcal{T}_{\mathbf{Q}}$
Additional	$O(k(m+n))$	$O(k)$	$O(k^2(m+n)+k^3))$

Here $\mathcal{T}_{\mathbf{A}}$ and $\mathcal{T}_{\mathbf{Q}}$ model the costs associated with computing matrix-vector products with \mathbf{A} and \mathbf{Q} (or $\partial_{\theta_i}\mathbf{Q}$) respectively

3.2 Computational cost

To derive an estimate of the computational cost, we have to make certain assumptions. First, assume that the computational cost associated with computing the forward operator or its adjoint, \mathbf{Ax} or $\mathbf{A}^T\mathbf{y}$ respectively (where $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^n \times \mathbb{R}^m$), can be modeled by $\mathcal{T}_{\mathbf{A}}$. Additionally, assume that the cost associated with computing \mathbf{Qx} or $\partial_{\theta_i}\mathbf{Qx}$ for any $1 \leq i \leq K$ is $\mathcal{T}_{\mathbf{Q}}$. Finally, we take \mathbf{R} to be a diagonal matrix which assumes that the noise is uncorrelated.

With these assumptions in place, the total cost associated with executing Algorithm 1 is

$$2(k+1)\mathcal{T}_{\mathbf{A}} + (2k+1)\mathcal{T}_{\mathbf{Q}} + O(k(m+n)) \text{ flops},$$

where “flops” refers to floating point operations. This represents the dominant cost of our approach.

Once the genGK matrices have been computed, it is straightforward to compute the objective function and the gradient. For the objective function, we need to compute the SVD of \mathbf{B}_k , so the additional cost of computing $\tilde{\mathcal{F}}_k$ is $O(k^3)$ flops. The cost of computing the gradient is additionally $K(k\mathcal{T}_{\mathbf{Q}} + O(k^2(m+n)+k^3))$ flops.

We summarize these results in Table 1.

3.3 Error analysis of genGK approximations

The following two propositions quantify the errors between \mathcal{F} and an approximation of \mathcal{F} when the approximation is generated by a low-rank representation of the matrix \mathbf{Z} . These results provide insight into the accuracy of the genGK process and help develop an *a posteriori* error estimator to monitor the error in \mathcal{F}_k . Let $\sigma_1(\widehat{\mathbf{A}}) \geq \sigma_2(\widehat{\mathbf{A}}) \geq \dots \geq \sigma_r(\widehat{\mathbf{A}}) > \sigma_{r+1}(\widehat{\mathbf{A}}) = \dots = \sigma_{\min\{m,n\}}(\widehat{\mathbf{A}}) = 0$ be the singular values of $\widehat{\mathbf{A}}$ and $r = \text{rank}(\widehat{\mathbf{A}})$. Let $\widehat{\mathbf{A}}_k = \mathbf{U}\Sigma_k\mathbf{V}^T$ be the k -rank approximation of the matrix $\widehat{\mathbf{A}} = \mathbf{R}^{-1/2}\mathbf{A}\mathbf{Q}^{1/2}$ for $1 \leq k < r$. The singular values of $\widehat{\mathbf{A}}$ are also the generalized singular values of the matrix \mathbf{A} in the sense of [17, Definition 3] with the matrix pair (\mathbf{R}, \mathbf{Q}) . The matrices $\mathbf{U} \in \mathbb{R}^{m \times k}$ and $\mathbf{V} \in \mathbb{R}^{n \times k}$ contain the left and right singular vectors respectively.

Proposition 3.1 Define the approximation to $\tilde{\mathcal{F}}$ using the truncated SVD of $\widehat{\mathbf{A}}$ as

$$\widehat{\mathcal{F}}_k = -\log \pi[\bar{\theta}] + \frac{1}{2}\log\det(\mathbf{Z}_k) + \frac{1}{2}\|\mathbf{A}\boldsymbol{\mu} - \mathbf{d}\|_{\mathbf{Z}_k^{-1}}^2, \quad (21)$$

with $\mathbf{Z}_k = \mathbf{R}^{1/2} [\widehat{\mathbf{A}}_k \widehat{\mathbf{A}}_k^\top + \mathbf{I}_m] \mathbf{R}^{1/2}$. Then

$$|\bar{\mathcal{F}} - \widehat{\mathcal{F}}_k| \leq \frac{1}{2} \sum_{i=k+1}^r \log(1 + \sigma_i^2(\widehat{\mathbf{A}})) + \frac{1}{2} \beta_1^2 \left(\frac{\sigma_{k+1}^2(\widehat{\mathbf{A}})}{1 + \sigma_{k+1}^2(\widehat{\mathbf{A}})} \right), \quad (22)$$

where $\beta_1 = \|\mathbf{A}\boldsymbol{\mu} - \mathbf{d}\|_{\mathbf{R}^{-1}}$.

Proof See Appendix B.1. \square

This proposition says that if the singular values of $\widehat{\mathbf{A}}$ (alternatively, generalized singular values of \mathbf{A}) decay rapidly, then the low-rank approximation $\widehat{\mathbf{A}}_k$ obtained using the truncated SVD results in an accurate approximation of the objective function. Indeed, if $\text{rank}(\widehat{\mathbf{A}}) = k$, then $\widehat{\mathcal{F}}_k = \mathcal{F}$. However, repeated computation of the truncated SVD of $\widehat{\mathbf{A}}$ at each optimization step is computationally expensive. As numerical experiments will demonstrate, the low-rank approximation obtained using genGK is accurate and is also computationally efficient. The next result derives an error estimate for the absolute error in the approximation of the objective function, where the approximation is computed using genGK.

Proposition 3.2 Consider $\bar{\mathcal{F}}$ and $\tilde{\mathcal{F}}_k$ as defined in (13) and (16) respectively. Define the orthogonal projector, $\Pi_{\mathbf{V}_k} \equiv \mathbf{Q}^{1/2} \mathbf{V}_k (\mathbf{Q}^{1/2} \mathbf{V}_k)^\top$ and let $\mathbf{H}_Q \equiv \mathbf{Q}^{1/2} \mathbf{H} \mathbf{Q}^{1/2}$ with $\mathbf{H} = \mathbf{A}^\top \mathbf{R}^{-1} \mathbf{A}$ and $\mathbf{H}_Q^{(k)} = \Pi_{\mathbf{V}_k} \mathbf{H}_Q \Pi_{\mathbf{V}_k}$. Then the absolute error in the objective function satisfies

$$|\bar{\mathcal{F}} - \tilde{\mathcal{F}}_k| \leq \frac{1}{2} \left[\xi_k + \beta_1^2 \left(\frac{\xi_k}{1 + \xi_k} \right) \right], \quad (23)$$

where $\xi_k = \text{tr}(\mathbf{H}_Q) - \text{tr}(\mathbf{H}_Q^{(k)})$, and $\beta_1 = \|\mathbf{d} - \mathbf{A}\boldsymbol{\mu}\|_{\mathbf{R}^{-1}}$.

Proof See Appendix B.2. \square

By [16], we have the following recurrence for $\xi_k = \text{tr}(\mathbf{H}_Q - \mathbf{H}_Q^{(k)})$. For any $k < \min\{m, n\} - 1$, ξ_k satisfies

$$\xi_{k+1} = \xi_k - (\alpha_{k+1}^2 + \beta_{k+2}^2).$$

This shows that the term ξ_k is monotonically nonincreasing. Therefore, Proposition 3.2 says that the absolute error can be bounded by a monotonically nonincreasing function. Furthermore, this proposition gives us a way to monitor the accuracy of the objective function and determine a suitable stopping criterion for terminating the iterations. This is discussed next.

3.4 Monitoring convergence of genGK

As mentioned earlier, an important issue that needs to be addressed is a stopping criterion to determine the value of k with which to approximate the objective function and the gradient.

Proposition 3.2 can be converted to a stopping criterion to monitor convergence, but the issue is that the initial iterate $\xi_0 = \text{tr}(\mathbf{H}_\mathbf{Q})$ cannot be computed efficiently in a matrix-free fashion. To address this issue we use a combination of the Monte Carlo trace estimator and the genGK recurrence to monitor ξ_k ; then in combination with the upper bound in Proposition 3.2 we can monitor the convergence.

Monte Carlo estimator

First, we describe the Monte Carlo estimators for the trace of a matrix. Let $\mathbf{K} \in \mathbb{R}^{n \times n}$ be a square matrix and suppose we are interested in estimating its trace using matrix-free techniques. Let $\boldsymbol{\omega} \in \mathbb{R}^n$ be a random vector with mean zero and identity covariance matrix. Then note that

$$\mathbb{E}[\boldsymbol{\omega}^\top \mathbf{K} \boldsymbol{\omega}] = \mathbb{E}[\text{tr}(\boldsymbol{\omega}^\top \mathbf{K} \boldsymbol{\omega})] = \text{tr}(\mathbf{K} \mathbb{E}[\boldsymbol{\omega} \boldsymbol{\omega}^\top]) = \text{tr}(\mathbf{K}),$$

where we have used the cyclic property of trace. Therefore, based on this identity, we can use a Monte Carlo estimator for $\text{tr}(\mathbf{K})$ as

$$\text{tr}(\mathbf{K}) \approx \frac{1}{n_{\text{mc}}} \sum_{j=1}^{n_{\text{mc}}} \boldsymbol{\omega}_j^\top \mathbf{K} \boldsymbol{\omega}_j = \frac{1}{n_{\text{mc}}} \text{tr}(\boldsymbol{\Omega}^\top \mathbf{K} \boldsymbol{\Omega}) \equiv \tilde{\text{tr}}(\mathbf{K}),$$

where $\boldsymbol{\omega}_j$ for $1 \leq j \leq n_{\text{mc}}$ are independent random vectors from the distribution of $\boldsymbol{\omega}$ and $\boldsymbol{\Omega} = [\boldsymbol{\omega}_1 \dots \boldsymbol{\omega}_{n_{\text{mc}}}]$.

This next result analyzes the error in the trace estimator applied to the matrix $\mathbf{H}_\mathbf{Q}$. Note that applying the Monte Carlo trace estimator to $\mathbf{H}_\mathbf{Q}$ only requires matvecs with \mathbf{H} and \mathbf{Q} ; in contrast, applying it to $\mathbf{H}_\mathbf{Q}$ is not practical since it requires matvecs with $\mathbf{Q}^{1/2}$.

Proposition 3.3 *Let $\boldsymbol{\omega}_k \in \mathbb{R}^n$ for $1 \leq k \leq N$ be random vectors with independent sub-Gaussian entries that have zero mean and $\max_j \|(\boldsymbol{\omega}_k)_j\|_{\psi_2} \leq K_\psi$ (see [18, Section 2.5.2] for a definition of the norm $\|\cdot\|_{\psi_2}$). The error in the trace estimator satisfies the probabilistic bound*

$$\mathbb{P}\{|\tilde{\text{tr}}(\mathbf{H}_\mathbf{Q}) - \text{tr}(\mathbf{H}_\mathbf{Q})| \geq t\} \leq 2 \exp\left(-C_{\text{HW}} N \min\left\{\frac{t^2}{K_\psi^4 \|\mathbf{H}_\mathbf{Q}\|_F^2}, \frac{t}{K_\psi^2 \|\mathbf{H}_\mathbf{Q}\|_2}\right\}\right), \quad (24)$$

where C_{HW} is an absolute constant. Furthermore, let $\epsilon > 0$ be the desired relative error and $0 < \delta < 1$ the desired failure probability. Then, with probability at least $1 - \delta$, the number of samples N_{samp} must satisfy

$$N_{\text{samp}} \geq \frac{K_\psi^2 \log(2/\delta)}{C_{\text{HW}} \epsilon^2} \left(\frac{K_\psi^2 \|\mathbf{H}_\mathbf{Q}\|_F^2}{\text{tr}(\mathbf{H}_\mathbf{Q})^2} + \frac{\epsilon \|\mathbf{H}_\mathbf{Q}\|_2}{\text{tr}(\mathbf{H}_\mathbf{Q})} \right)$$

to ensure $|\tilde{\text{tr}}(\mathbf{H}_\mathbf{Q}) - \text{tr}(\mathbf{H}_\mathbf{Q})| \leq \epsilon |\text{tr}(\mathbf{H}_\mathbf{Q})|$.

Proof See Appendix C. □

The novelty in this result is that it does not require \mathbf{K} to be symmetric or positive semidefinite as is the standard assumption (see, e.g., [19] and references within).

In practice, we take $\boldsymbol{\Omega}$ to be a standard Gaussian random matrix, i.e., with independent and identically distributed entries drawn from $\mathcal{N}(0, 1)$. Another option is to choose the entries of the random matrix to be independent Rademacher random variables (entries ± 1 with equal probability). The analysis in Proposition 3.3 applies to both distributions since both distributions are sub-Gaussian with sub-Gaussian norm $K_\psi = \mathcal{O}(1)$.

Algorithm 3 Monitoring accuracy of the genGK approximations

Require: Matrices $\mathbf{H}, \mathbf{Q}, \mathbf{V}_k, \mathbf{T}_k$. Integers k_{\max}, n_{mc} .

- 1: Draw $\boldsymbol{\Omega} \in \mathbb{R}^{n \times n_{\text{mc}}}$ with i.i.d. entries from $\mathcal{N}(0, 1)$
- 2: Compute $\mathbf{Y} = \mathbf{HQ}\boldsymbol{\Omega}$
- 3: **for** $k = 1, \dots, k_{\max}$ **do**
- 4: Compute $\mathbf{Y}_k = \mathbf{Y} - \mathbf{V}_k \mathbf{T}_k \mathbf{V}_k^\top \mathbf{Q}\boldsymbol{\Omega}$
- 5: Estimate $\tilde{\xi}_k = \text{tr}(\boldsymbol{\Omega}^\top \mathbf{Y}_k) / n_{\text{mc}}$
- 6: **end for**
- 7: **return** Estimate $(\tilde{\xi}_k)_{k=1}^{k_{\max}}$.

Monitoring error

Next, we explain how to estimate ξ_k using the Monte Carlo trace estimator. First consider ξ_k , which can be written using the cyclic property of the trace as

$$\xi_k = \text{tr}(\mathbf{H}\mathbf{Q}) - \text{tr}(\mathbf{H}\mathbf{Q}^{(k)}) = \text{tr}(\mathbf{HQ} - \mathbf{V}_k \mathbf{T}_k \mathbf{V}_k^\top \mathbf{Q}).$$

Note that in this formulation, we do not need to work with the square roots of \mathbf{Q} . Applying the Monte Carlo trace estimator with $\boldsymbol{\Omega}$ as a standard Gaussian random matrix, we can estimate ξ_k as

$$\tilde{\xi}_k := \frac{1}{n_{\text{mc}}} \text{tr}(\boldsymbol{\Omega}^\top (\mathbf{HQ} - \mathbf{V}_k \mathbf{T}_k \mathbf{V}_k^\top \mathbf{Q}) \boldsymbol{\Omega}).$$

Note that this algorithm only requires matrix-vector products with \mathbf{H} and \mathbf{Q} and the genGK relationships. The details and efficient implementation of this estimator are given in Algorithm 3. In practice, we take the number of Monte Carlo samples $n_{\text{mc}} = 10$. Finally, we use the estimator

$$\text{err}_{\text{mc}} := \frac{1}{2} \left[\tilde{\xi}_k + \beta_1^2 \frac{\tilde{\xi}_k}{1 + \tilde{\xi}_k} \right],$$

to estimate the error in the objective function. Note that while the upper bound in Proposition 3.2 no longer holds with the Monte Carlo estimators, it can still be used to monitor the accuracy as an error indicator.

4 Numerical experiments

In the following, we outline experiments corresponding to problems in one and two dimensions that arise in inverse problems. Using our approach, we demonstrate that approximations to both the objective function \mathcal{F} and its gradient $\nabla\mathcal{F}$ using the genGK relations can lead to hyperparameter estimates that result in accurate reconstructions. We also show that one can effectively inform the selection of the bidiagonalization parameter k using the error bounds derived above. Tests of robustness are presented, and for the one-dimensional case, illustrations for the computational need for a framework such as ours are provided.

All experiments are performed using MATLAB, with optimizations done via `fmincon` with an interior point method [20–22]. A MATLAB implementation of the algorithms can be found in https://github.com/Inverse-Modeling/EB_genGK.

Choice of priors and hyperpriors

Although our approach is general, for concreteness we assume that the dimensionality of our hyperparameter space is three (i.e., $K = 3$). The first hyperparameter controls the variance of the noise. We assume that the noise is Gaussian, with zero mean and covariance $\mathbf{R}(\boldsymbol{\theta}) = \theta_1 \mathbf{I}_m$. We model the prior covariance matrix using the Matérn covariance family. This is a flexible family of covariance kernels that can be used to model a wide range of behaviors with a relatively few set of parameters. The kernel is isotropic and is defined using the covariance function

$$\mathcal{M}_{v, \sigma^2, \ell}(r) \equiv \frac{\sigma^2}{2^{v-1} \Gamma(v)} \left(\sqrt{2v} \frac{r}{\ell} \right)^v \mathcal{K}_v \left(\sqrt{2v} \frac{r}{\ell} \right)$$

where Γ is the gamma function, \mathcal{K}_v is the modified Bessel function of the second kind, and the positive parameters v , σ^2 , ℓ represent the smoothness of the process, the prior variance, and the correlation length respectively. Given the covariance function $\mathcal{M}_{v, \sigma^2, \ell}(r)$, the covariance kernel is given by $\kappa(\mathbf{x}, \mathbf{y}) = \mathcal{M}_{v, \sigma^2, \ell}(\|\mathbf{x} - \mathbf{y}\|_2)$. Given a set of points $\{\mathbf{x}_j\}_{j=1}^n$ the covariance matrix $\mathbf{Q}(\boldsymbol{\theta})$ can be constructed entrywise as

$$[\mathbf{Q}(\boldsymbol{\theta})]_{i,j} = \mathcal{M}_{v, \theta_2^2, \theta_3}(\|\mathbf{x}_i - \mathbf{x}_j\|_2) \quad 1 \leq i, j \leq n.$$

Therefore, θ_2 and θ_3 represent the prior standard deviation and the correlation length. We do not estimate the smoothness v as a part of the estimation process but assume that it is fixed. We also assume the prior mean $\boldsymbol{\mu}(\boldsymbol{\theta}) = \mathbf{0}$. In all the numerical experiments, we use the FFT-based technique to compute matvecs in $\mathcal{O}(n \log n)$ [9].

For the hyperpriors, following [5], we report experiments with two different choices: (P1) is the improper prior chosen as $\pi(\boldsymbol{\theta}) \propto 1$; and (P2) is a Gamma prior with

$$\pi(\boldsymbol{\theta}) \propto \exp \left(- \sum_{i=1}^K \gamma \theta_i \right) \quad \theta_i > 0, \quad 1 \leq i \leq K. \quad (25)$$

The parameter γ is set to be 10^{-4} and chosen such that the probability density function is relatively flat over the parameter space. We experimented with other hyperpriors and obtained similar results.

Experimental setup

To generate the noise we adopt the following procedure. A linear forward operator \mathbf{A} along with some exact true signal \mathbf{s} is constructed so that the uncorrupted data, \mathbf{d}_* , is $\mathbf{d}_* = \mathbf{A}\mathbf{s}$. Then we generate a realization, $\boldsymbol{\epsilon}$, of a Gaussian random vector $X_{\boldsymbol{\epsilon}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_m)$. Next, we set

$$\boldsymbol{\eta} = \boldsymbol{\epsilon} \frac{\lambda_{\text{noise}} \|\mathbf{d}_*\|_2}{\|\boldsymbol{\epsilon}\|_2},$$

where $\lambda_{\text{noise}} \in (0, \infty)$ represents a noise level parameter; by construction, it is such that $\|\boldsymbol{\eta}\|_2 = \lambda_{\text{noise}} \|\mathbf{d}_*\|_2$. Finally, we generate a vector of noisy data observations via $\mathbf{d} = \mathbf{d}_* + \boldsymbol{\eta}$.

We compare the accuracy in terms of relative reconstruction error norms, defined as

$$\text{RE} = \frac{\|\mathbf{s} - \tilde{\mathbf{s}}\|_2}{\|\mathbf{s}\|_2},$$

where \mathbf{s} is the true solution and $\tilde{\mathbf{s}}$ is the approximation. The timing computations were performed on North Carolina State University's High Performance Computing cluster 'Hazel' using MATLAB R2023a. Specifically, our computing resource utilized an Intel Xeon Gold 6226 microprocessor with 188 GB of RAM, 2 sockets, and 32 cores per socket.

4.1 Application 1: inverse heat transfer

Our first experiment corresponds to a one-dimensional problem in heat conduction, the details of which are outlined in [23–27]. Let $L, \kappa > 0$, and define

$$K : [0, L] \times [0, L] \rightarrow \mathbb{R} : (t, s) \mapsto \frac{1}{\sqrt{4\pi\kappa^2}} (s - t)^{-3/2} \exp\left(-\frac{1}{4}(s - t)^{-1}\right), \quad (26)$$

and for any $t \in [0, L]$, define the integral operator

$$(T\varphi)(t) = \int_0^t K(t, s)\varphi(s) ds. \quad (27)$$

with the parameter $\kappa = 1$. The parameter κ controls the degree of ill-posedness of the problem; here, $\kappa = 1$ yields an ill-posed problem, whereas $\kappa = 5$ gives a well-posed problem. The inverse problem consists of determining a function f such that $Tf = g$, where g is known.

Using the Regularization Tools package [24], a discretization of the operator T , in the domain $\Omega = [0, 1]$, is generated. We denote the discretized representation of T

as $\mathbf{A} \in \mathbb{R}^{n \times n}$ and the approximation s for the function f . For the prior covariance, we choose the Matérn covariance with $\nu = 3/2$ and for the hyperpriors, we used the improper prior (P1). The measurements were corrupted with 2% additive Gaussian noise.

Experiment 1: Accuracy of objective function

In this experiment, we investigate the accuracy of the genGK estimates in computing the objective function at the optimal value $\boldsymbol{\theta}^*$ (this will be discussed in Experiment 2). In the left plot of Fig. 1, we plot the relative error of $\tilde{\mathcal{F}}_k$ for various k (blue, solid line) and compare it against the estimated bound derived using Monte Carlo techniques with $n_{mc} = 10$ (red, dashed line).

We see from Fig. 1 (left panel) that the computable bound reasonably tracks the observed errors over a large range of k values. To investigate further, we separated the main contributions to the objective function: the log-determinant term $\frac{1}{2}\log\det(\mathbf{Z}(\boldsymbol{\theta}))$ and the quadratic term $\frac{1}{2}\|\mathbf{A}\boldsymbol{\mu}(\boldsymbol{\theta}) - \mathbf{d}\|_{\mathbf{Z}(\boldsymbol{\theta})^{-1}}^2$. The relative errors in these terms are defined as

$$RE_{\logdet} = \frac{|\log\det(\mathbf{Z}) - \log\det(\tilde{\mathbf{Z}}_k)|}{|\log\det(\mathbf{Z})|} \quad RE_{\text{quad}} = \frac{\|\mathbf{A}\boldsymbol{\mu} - \mathbf{d}\|_{\mathbf{Z}^{-1}}^2 - \|\mathbf{A}\boldsymbol{\mu} - \mathbf{d}\|_{\tilde{\mathbf{Z}}_k^{-1}}^2}{\|\mathbf{A}\boldsymbol{\mu} - \mathbf{d}\|_{\mathbf{Z}^{-1}}^2}.$$

In the above expressions, we have suppressed the dependence on $\boldsymbol{\theta}$. Correspondingly the appropriate computable bounds are $CB_{\logdet} = \frac{\hat{\xi}_k}{2\mathcal{F}}$ and $CB_{\text{quad}} = \frac{\beta_1^2 \hat{\xi}_k}{2\mathcal{F}(1 + \hat{\xi}_k)}$; see Section 3.4. From the middle and right plots of Fig. 1, we observe that the bound for the log determinant term is better than for the quadratic term. This is explored further in Experiment 2.

Moreover, we observe that the errors in the objective function approximations exhibit sharp decay with increasing values of k . To explain this behavior, we plot the generalized singular values of the operator. We can observe that the rank- k SVD approximation produces comparable results to the genGK approximation. Note that the errors using these approaches are not guaranteed to be monotonically decreasing. Nevertheless, the sharp decay in the error can be understood by considering the decay in the generalized singular values in Fig. 2.

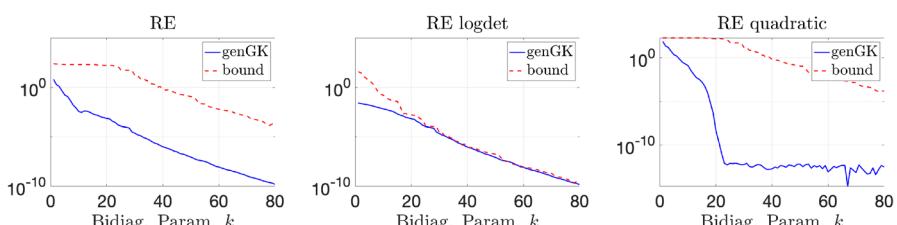


Fig. 1 1D Heat: Relative errors and corresponding bounds. The left panel compares the overall accuracy of the objective function; the middle and right panels compare the errors of two components of the objective function: the log determinant and the quadratic terms respectively

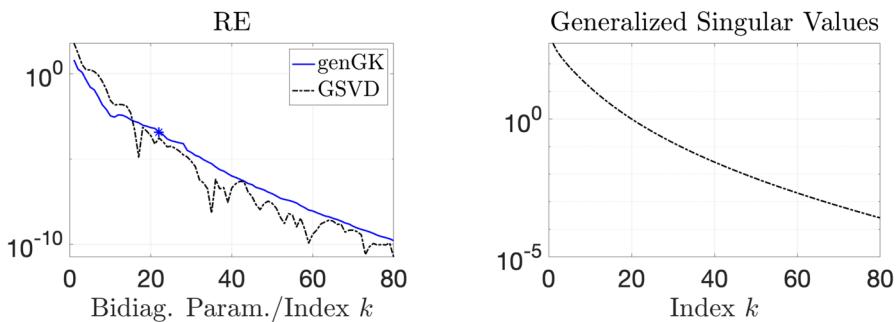


Fig. 2 1D Heat: (left) Relative error in the objective function using genGK and GSVD, and (right) plot of singular values of $\hat{\mathbf{A}}$

The blue star in the left plot denotes the k value that is used in our subsequent numerical experiments ($k = 22$). The relative error of the objective function at this point is approximately 10^{-4} , with the relative error in the quadratic term being approximately 10^{-11} . An empirical justification for this choice is illustrated in the following sections.

Experiment 2: Recovery and accuracy along optimization trajectory

In this experiment, we discuss the performance of the optimization solver, the recovery, and the accuracy of the objective function along the optimization trajectory. We use the same setup as before and fix the number of genGK iterations to $k = 22$.

In Fig. 3(a), we plot the data without noise and with added noise of 2%. In Fig. 3(b), we plot a reconstruction alongside the true solution. The optimizer took 29 iterations to converge with 69 function and gradient evaluations (i.e., `funcCount = 69`) and the optimal solution was found to be

$$\boldsymbol{\theta}^* = (8.73 \times 10^{-7}, 0.2562, 0.0566)^\top.$$

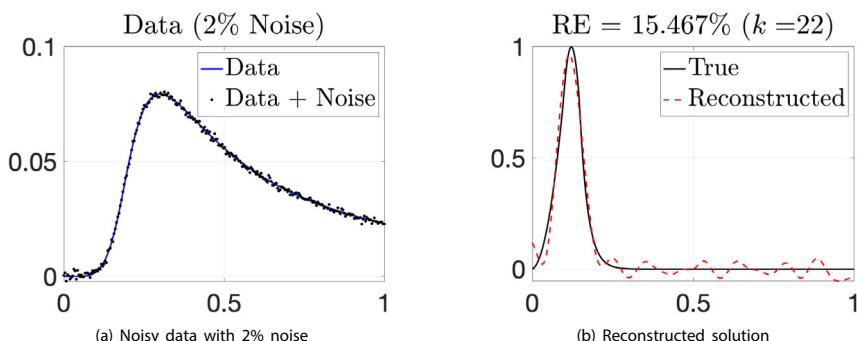


Fig. 3 1D Heat: On the left, we provide the observations (with and without noise). On the right, we provide the reconstruction, along with the true solution

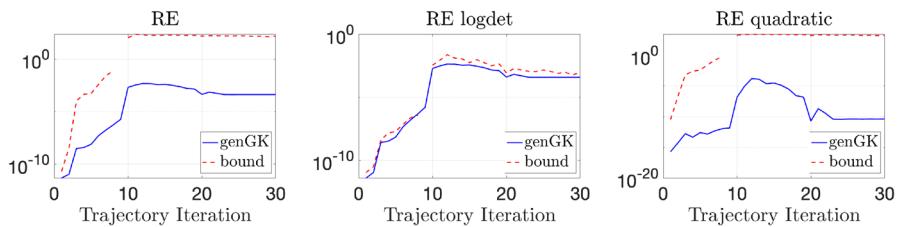


Fig. 4 1D Heat: Relative errors and corresponding bounds along the optimization trajectory for $k = 22$ and $n_{mc} = 10$. The left panel compares the overall accuracy in the objective function, and the middle and right panels compare the errors of the log determinant and the quadratic term respectively

The relative reconstruction error is found to be about 15.46% at the optimal value of θ^* .

Next, we investigate the accuracy of the objective function along the optimization trajectory and provide relative errors in Fig. 4. We first observe that the relative error in the objective function shows a steady increase until about 10 iterations after which it plateaus. The computable bound tracks the error well until about 10 iterations after which it shows poor quantitative behavior, although appears to be qualitatively good. We plot in the middle and right panels the error in the log determinant and quadratic terms respectively, with computable bounds provided. We observe that the log-determinant term dominates the error and shows similar behavior as the overall error. Also, the bound for the log-determinant term is quantitatively informative, whereas the bound for the quadratic term is poor, which explains why the overall bound is poor close to the optimal solution. We conclude by emphasizing two observations: the overall error estimate is reasonable, and the bound for the log-determinant term is reasonable, so this approach can be used as an error indicator.

In Fig. 5, we consider the error in the gradients. In the left plot, we plot the relative error (in the ∞ norm) in the gradient of the objective function \mathcal{F} , the gradient of the log-determinant term $\frac{1}{2}\logdet(\mathbf{Z})$, and the gradient of the quadratic term $\frac{1}{2}\|\mathbf{A}\boldsymbol{\mu} - \mathbf{b}\|_{\mathbf{Z}^{-1}}^2$. As in the case of the error in the objective function, the error in the gradient increases close to the optimal solution and the relative error in the gradient of the quadratic term

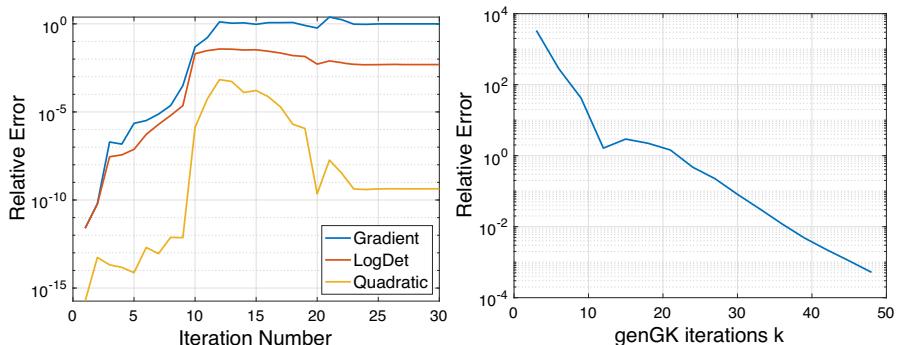


Fig. 5 1D Heat: Relative errors of the gradient. The left panel compares the overall accuracy in the gradient and its individual contributions from the log-determinant and quadratic terms, and the right panel compares the error with the number of genGK iterations k

is smaller than the error in the gradient of the log-determinant term. Note here, that the number of iterations k for genGK is kept at 22. In the right panel, we also plot the relative error in the gradient with increasing k at the optimal point θ^* . Like the objective function, the relative error exhibits fast decay with k .

We also investigated the robustness of the optimization procedure by choosing 100 different values of the initial guess θ_0 . The initial guesses are generated randomly by perturbing each coordinate uniformly at random by 50% of the optimal parameter. Once the initial guesses are determined, the optimization problem is solved for both the “exact” case (using \mathcal{F}) and the approximate case (using $\tilde{\mathcal{F}}_k$) with the value $k = 22$. We find that the optimization solutions converge to the same optimal solution with the relative error in the resulting images between 14 – 15% thus demonstrating the robustness of the genGK approximations.

Experiment 3: Computational time

To demonstrate the computational efficiency of our approach, we compute the CPU time required to compute the objective function and the gradient with increasing numbers of unknowns n . In Fig. 6, we provide the wall clock time needed to compute the objective function-gradient pair for both the exact and approximate cases. For the case $n = 8192$, the genGK approximation provides an approximation faster than the “full” case by a factor of 81.

4.2 Application 2: seismic tomography

This experiment corresponds to a two-dimensional problem in seismic travel-time tomography. Typically, this class of problems simulates geophysical situations in which measurements of the travel time of seismic waves are recorded between a collection of sources and detectors. Utilizing these measurements, one can recon-

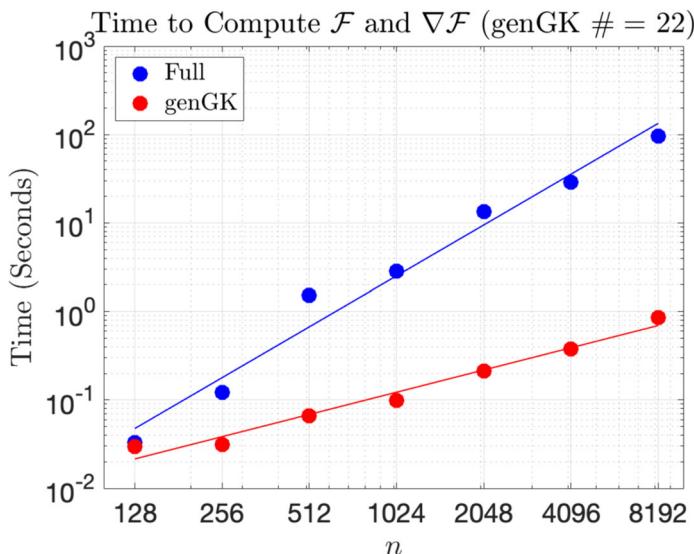


Fig. 6 1D Heat: Wall clock time to compute the objective function and gradient pair versus the problem size n

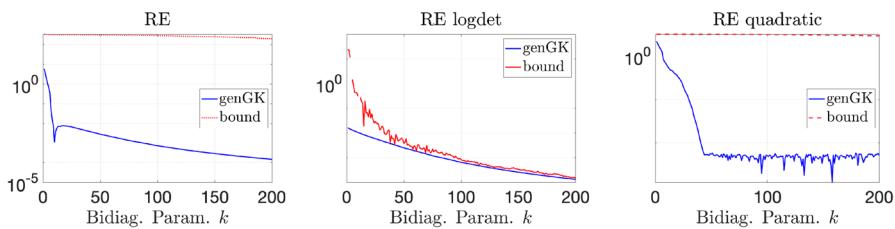


Fig. 7 Seismic tomography: Relative errors and corresponding bounds for the objective function (left), log determinant term (middle), and quadratic term (right)

struct an image associated with the waves in some specified domain $\Omega = [0, 1]^2$. We use the IR Tools [28] package to generate the test problem. Using the `[A, d, s, info] = PRseismic(n, options)` command from the package, we obtain a forward operator, \mathbf{A} , a true solution, \mathbf{s} , and a right-hand-side of true observations, \mathbf{d} . For the experiments below (via the input options), we use the following settings: `phantomImage = 'smooth'`, `wavemodel = 'ray'`, $s = 32$, and $p = 45$. This setup yields a forward operator with 1440 measurements. As before, 2% noise is added to the data to simulate measurement error. Additionally, we choose the hyperprior (P2), $\pi[\boldsymbol{\theta}] \propto \exp\left(-\gamma \sum_{i=1}^3 \theta_i\right)$ with $\gamma = 10^{-4}$, and the covariance matrix is constructed using the Matérn kernel with $\nu = 3/2$. We begin with a similar set of experiments to the first application. We take the number of unknowns to be 4096 since it is easier to compare the accuracy against the true objective function for a smaller problem. Contrary to application 1, this application is a significantly under-determined system. Moreover, in Section 4.3 we see how a 2-hyperparameter setting can be used for this problem.

Experiment 1: Accuracy of objective function

In Fig. 7, we plot the relative error of the objective function at a point $\boldsymbol{\theta}^*$ in blue and its estimated bound in red with $n_{mc} = 10$. Similar to the 1D Heat example and as suggested by the decay of the singular values in Fig. 8, the relative errors exhibit a sharp decay with k ; however, we see that the bound, while qualitatively good, is not accurate. By examining separately the two components, we observe that the bound for the log-determinant term is quite good while the bound for the quadratic term is

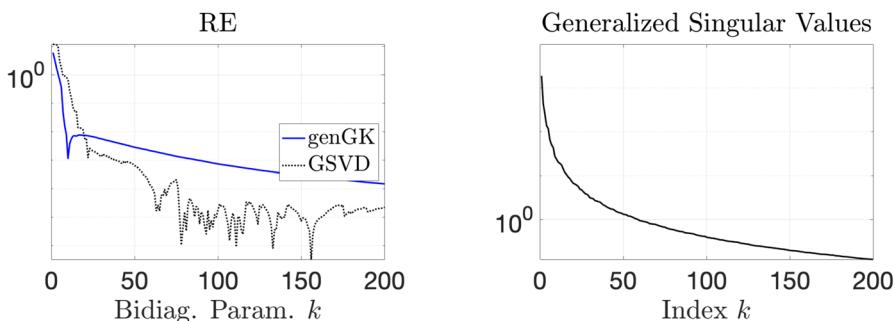


Fig. 8 Seismic tomography: (left) Relative error in the objective function using genGK and GSVD, and (right) plot of singular values of $\widehat{\mathbf{A}}$

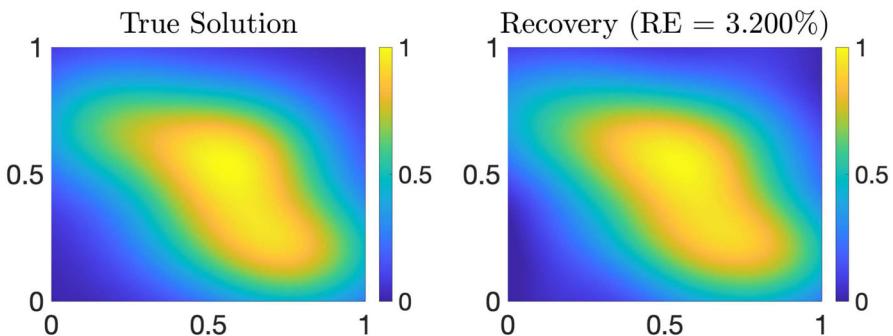


Fig. 9 Seismic tomography: True solution and reconstruction obtained with computed hyperparameters

not great. In subsequent numerical experiments, we used $k = 200$ at which point the error in the objective function is about 10^{-5} .

Experiment 2: Reconstruction and accuracy along optimal trajectories

The optimization procedure for estimating the hyperparameters took 23 iterations and the number of objective function evaluations was 61. The true and reconstructed solutions are provided in Fig. 9. The recovered image has about 3% relative reconstruction error and similar qualitative features to the true solution.

The previous experiment only considered approximations at a single set of parameter values for θ . In Fig. 10 we plot the relative errors of the objective function as well as the two components, log-determinant and quadratic, along the optimization trajectory. As before, we see that the relative error is consistently good throughout the optimization trajectory. Looking more closely, we see that the relative error in the quadratic component is much smaller than the log determinant term (in fact, it is close to machine precision), suggesting that the genGK approximation is much better at approximating the quadratic term. Now considering the bounds, we see that the bound for the log determinant is very good both qualitatively and quantitatively, and as before the bound for the quadratic term is very poor. Therefore, we can use the bound for the log determinant as an indicator for the error in the objective function and the gradient.

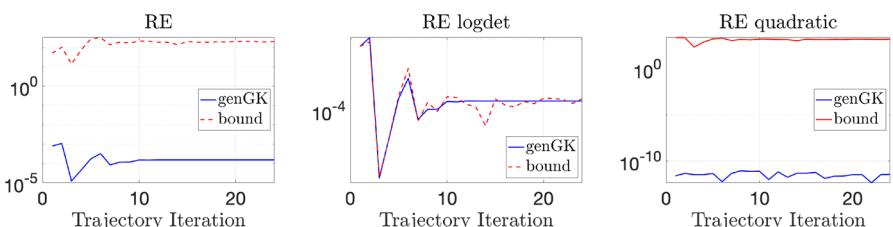
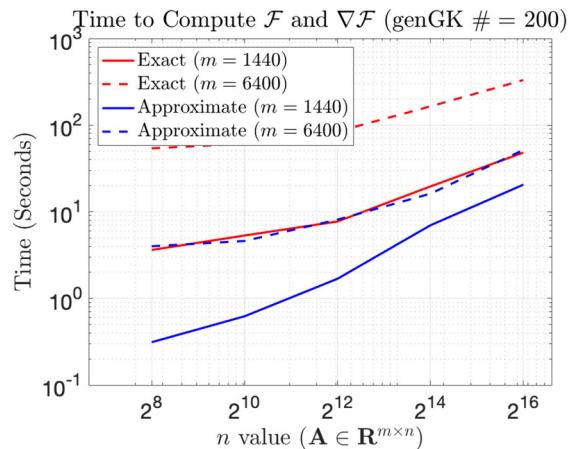


Fig. 10 Seismic tomography: Relative errors and corresponding bounds along the optimal trajectory for $k = 200$ and $n_{mc} = 100$ samples. The left panel compares the overall accuracy of the objective function, and the middle and right panels compare the errors of the log determinant and the quadratic term respectively

Fig. 11 Seismic tomography: Wall clock time to compute the objective function and gradient for two different values of m : 1440 and 6400



Experiment 3: Computational time

In Fig. 11 we compare the computational time of our approach against the full approach. We choose two values of m , namely 1440 and 6400 (with $s=64$ and $p=100$). As before, we present the time to compute the objective function and the gradient for various numbers of unknowns n from 2^8 to 2^{16} . We see that for both sets of measurements, the proposed approach that uses genGK (“Approximate”) is much faster than the full (“Exact”) approach. The difference is more pronounced for a larger number of measurements $m = 6400$.

4.3 Two-hyperparameter setting

In the previous examples, the parameter θ_1 is related to the variance of the noise, the parameter θ_2 is related to the variance of the prior, and θ_3 is the correlation length of the prior covariance. Consider the setting where $\eta \sim \mathcal{N}(\mathbf{0}, \theta_1 \mathbf{I}_m)$, $s \sim \mathcal{N}(\mu, \theta_2^2 \mathbf{Q}_0)$, and $\mathbf{Q}_0 \in \mathbb{R}^{n \times n}$. In this setting, one has access to a reliable estimate for θ_3 , resulting in a fixed prior covariance matrix \mathbf{Q}_0 and wishes to simply optimize over the parameters $\theta = (\theta_1, \theta_2) \in \mathbb{R}_+^2$. At first glance, it appears to be a special case of our approach, but this special case is of interest for two reasons: first, it allows us to compare against existing parameter selection methods that can recover a ratio of these parameters; and second, the computational cost of the optimization problem can be substantially lowered since the genGK relations can be precomputed in an offline stage. We elaborate on both points below.

Table 2 Comparison of the regularization parameter λ or $1/\theta_2$ after optimizing over a two-dimensional hyperparameter space

	Our Approach	“Optimal”	WGCV	DP
θ_2 or $1/\lambda$	0.49	0.46	0.09	0.27
Rel. Err.	2.48%	2.44%	4.59%	3.41%
Iterations (MAP estimate)	150	43	58	18

To explain the first point, in [1] the genHyBR approach was considered for computing the MAP estimate

$$\min_{\mathbf{s} \in \mathbb{R}^n} \|\mathbf{d} - \mathbf{A}\mathbf{s}\|_{\mathbf{R}^{-1}}^2 + \lambda^2 \|\mathbf{s} - \boldsymbol{\mu}\|_{\mathbf{Q}_0^{-1}}^2,$$

with methods to automatically estimate the regularization parameter λ , e.g., using the Generalized Cross Validation (GCV) method, the Unbiased Predictive Risk Estimate (UPRE), the Discrepancy Principle, etc. The connection between this formulation and the present approach is that $\lambda^2 = 1/\theta_2^2$. A key distinction of our approach presented here is that we can recover both θ_1 and θ_2 , whereas genHyBR can recover only the ratio $\sqrt{\theta_1}/\theta_2$ (assuming that θ_1 is known a priori). We will present a comparison with genHyBR below.

To expand on the second point, we argue that the genGK process needs to be run only once, and the computed matrices can be reused during the optimization process. That is, we run Algorithm 1 as genGK(\mathbf{A} , \mathbf{I} , \mathbf{Q}_0 , $\boldsymbol{\mu}$, \mathbf{d} , k) to obtain the matrices $\widehat{\mathbf{U}}_{k+1}$, $\widehat{\mathbf{B}}_k$, and $\widehat{\mathbf{V}}_k$ that satisfy $\mathbf{A} \approx \widehat{\mathbf{U}}_{k+1} \widehat{\mathbf{B}}_k \widehat{\mathbf{V}}_k^\top$ with the orthogonality relations $\widehat{\mathbf{U}}_{k+1}^\top \mathbf{U}_{k+1} = \mathbf{I}_{k+1}$ and $\widehat{\mathbf{V}}_k^\top \mathbf{Q}_0 \widehat{\mathbf{V}}_k = \mathbf{I}_k$. To obtain the genGK relations for any particular value of $\boldsymbol{\theta} = (\theta_1, \theta_2)$ one simply has to perform the change of variables

$$\mathbf{U}_{k+1} = \sqrt{\theta_1} \widehat{\mathbf{U}}_{k+1}, \quad \mathbf{V}_k = \theta_2^{-1} \widehat{\mathbf{V}}_k, \quad \mathbf{B}_k = \frac{\theta_2}{\sqrt{\theta_1}} \widehat{\mathbf{B}}_k.$$

This gives the genGK iterations at the new optimization point that satisfies $\mathbf{A} \approx \widehat{\mathbf{U}}_{k+1} \widehat{\mathbf{B}}_k \widehat{\mathbf{V}}_k^\top = \mathbf{U}_{k+1} \mathbf{B}_k \mathbf{V}_k^\top$ with the new orthogonality relations

$$\mathbf{U}_{k+1}^\top (\theta_1 \mathbf{I}_m)^{-1} \mathbf{U}_{k+1} = \mathbf{I}_{k+1} \quad \mathbf{V}_k^\top (\theta_2^2 \mathbf{Q}_0) \mathbf{V}_k = \mathbf{I}_k.$$

These new genGK relations can be used in conjunction with the approximations developed in Section 3.1. More specifically, the computational cost at each iteration to determine the objective function and the gradient is $\mathcal{O}(k(m + n) + k^3)$ flops. The important point is that each iteration does not require applying the forward operator, and the optimization procedure is very efficient since the (expensive) genGK step is precomputed.

Application to Seismic Tomography

Consider the seismic tomography problem (Section 4.2) with the same settings except we take $\mathbf{A} \in \mathbb{R}^{1440 \times 65536}$ and the number of genGK iterations to be 150. The value of θ_3 was chosen based on the optimal value from the previous experiments and was set to $\theta_3^* = 0.90216$. Using our approach we obtained the optimal value $\boldsymbol{\theta}_* = (1.07 \times 10^{-4}, 0.49)^\top$. Additionally, in Table 2, we provide the values of the regularization parameter computed using genHyBR for comparison [1]. Since genHyBR can only estimate the parameter λ , rather than θ_1 and θ_2 separately, we fixed the value of the noise variance to be $\theta_1 = 1.07 \times 10^{-4}$. Then, we compute at each iteration the optimal regularization parameter λ_{opt} and denoted “Optimal”, which minimizes the 2-norm of the error between the reconstruction and the truth. We also consider

the parameter selection techniques using the weighted generalized cross-validation approach (WGCV) and the discrepancy principle (DP). The details are given in [1].

Table 2 lists the value of θ_2 (which is also the inverse regularization parameter) obtained using the different techniques. Our approach which optimizes over θ_1 and θ_2 has a comparable relative error to the “Optimal” approach which assumes knowledge of θ_1 and the true solution. Our approach also does better than WGCV and DP approaches in terms of relative error. In summary, the proposed approach is more general and compares favorably against existing special-purpose parameter selection techniques.

4.4 Atmospheric tomography

In this last numerical experiment, we illustrate our methodology on an atmospheric inverse modeling example [29], where the data vector \mathbf{d} represents noisy satellite observations of a hypothetical atmospheric trace gas. The matrix \mathbf{A} represents a forward atmospheric model that will transport this gas based on estimated atmospheric winds and simulate satellite observations. Specifically, \mathbf{A} is generated from atmospheric model runs conducted as part of NOAA’s CarbonTracker-Lagrange Project [30, 31]. The true solution is generated from a truncated Karhunen-Loéve expansion with the Matérn covariance kernel with $\nu = 5/2$, $\ell = 0.05$ and 20 terms in the expansion. An illustration of the true solution is given in the left panel of Fig. 12.

We generate a synthetic atmospheric transport problem where $\mathbf{A} \in \mathbb{R}^{98880 \times 3222}$ and $\mathbf{d} \in \mathbb{R}^{98880}$, and the goal is to reconstruct the unknown set of states or trace gas fluxes across North America, where the spatial resolution is $1^\circ \times 1^\circ$. Although the unknown parameters are over land, to exploit the advantages of the fast Fourier transform, we treat the unknowns on a regular grid and then incorporate a masking operator in the definition of \mathbf{A} . More specifically, the “unknown” vector \mathbf{s} is provided in Fig. 12(a) and is an image generated using a truncated Karhunen-Loéve expansion generated with a Matérn kernel with parameters $\nu = 2.5$ and $\ell = 0.05$ followed by a land mask. The observed data \mathbf{d} were obtained by applying the atmospheric forward model \mathbf{A} and corrupting it with 2% noise to simulate measurement noise. Note that in

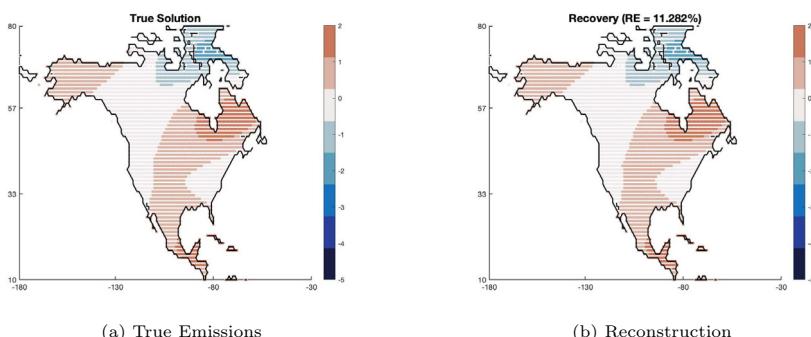


Fig. 12 Atmospheric tomography: (left) True synthetic emissions pattern, and (right) reconstruction with estimated hyperparameters

actual applications, the noise level is much higher (e.g., 50%). For additional details, we refer the reader to [29, 30, 32].

For the hyperparameter estimation and reconstruction, we take \mathbf{Q} defined by a Matérn kernel with $\nu = 3/2$. For the hyperprior, we used a noninformative prior (P1), and for genGK, we used $k = 250$ iterations. For the initial guess, we used an estimate of the noise variance θ_1 and the length scale $\theta_3 = 0.075$ (the length scale is based on a domain of length 1; we used a random guess for θ_2 . The optimizer took 19 iterations and 94 number of function evaluations. The optimal value of θ was found to be

$$\theta^* = (8.51 \times 10^{-5}, 1.068, 0.0135).$$

The reconstruction is provided in Fig. 12(b), where the resulting relative reconstruction error norm was approximately 11.2%. The optimization procedure took about 54 minutes in wall clock time.

5 Conclusions

This paper describes an efficient approach for hyperparameter estimation that combines an empirical Bayes approach with the generalized Golub-Kahan bidiagonalization process for large-scale linear inverse problems. We consider the marginal posterior distribution and derive efficient algorithms for computing the MAP estimate of the marginalized posterior. We derive an approximation to the objective function and its gradient, where the approximation is computed using information from the genGK bidiagonalization, and several results quantifying the accuracy of the approximations are provided. We also have a method to estimate and track the error in the approximations. We demonstrated the performance of our approach on model problems from inverse heat equation (1D), seismic and atmospheric tomography (2D). While our approach is fairly general, in numerical experiments we only considered the Matérn covariance models with a small number of hyperparameters. Similarly, while we only considered the empirical Bayes method here, the approximations that we derive are useful in other formulations such as fully Bayesian and variational Bayes.

In future work, we will consider more robust methods for estimating the hyperparameters that do not rely solely on low-rank approximations of \mathbf{A} . Another aspect worth exploring is whether we can improve the bound for the quadratic term. We will also consider extensions to more realistic time-dependent inverse problems, such as from atmospheric and dynamic tomography problems, where low-rank structures or sparsity patterns in the temporal direction may be exploited. Such problems come with additional computational challenges, including more hyperparameters to estimate and larger numbers of unknowns (e.g., corresponding to finer spatiotemporal discretizations).

Appendix A: Details of a derivation

We show how to derive (20). Using the Sherman–Morrison–Woodbury formula, we have $\tilde{\mathbf{Z}}_k^{-1} = (\mathbf{U}_{k+1}\mathbf{B}_k\mathbf{B}_k^\top\mathbf{U}_{k+1}^\top + \mathbf{R})^{-1} = \mathbf{R}^{-1} - \mathbf{R}^{-1}\mathbf{U}_{k+1}\mathbf{B}_k(\mathbf{I}_k + \mathbf{B}_k^\top\mathbf{B}_k)^{-1}\mathbf{B}_k^\top\mathbf{U}_{k+1}^\top\mathbf{R}^{-1}$.

Plugging in the expression for $\tilde{\mathbf{Z}}_k^{-1}$ and using the linearity of trace

$$\begin{aligned} \langle \tilde{\mathbf{Z}}_k^{-1}, \widetilde{\partial_{\theta_i} \mathbf{Z}_k} \rangle_F &= \text{tr}(\tilde{\mathbf{Z}}_k^{-1} \widetilde{\partial_{\theta_i} \mathbf{Z}_k}) \\ &= \text{tr}\left(\left[\mathbf{R}^{-1} - \mathbf{R}^{-1}\mathbf{U}_{k+1}\mathbf{B}_k(\mathbf{I}_k + \mathbf{T}_k)^{-1}\mathbf{B}_k^\top\mathbf{U}_{k+1}^\top\mathbf{R}^{-1}\right]\left[\widetilde{\mathbf{A}}_k \partial_{\theta_i} \mathbf{Q} \widetilde{\mathbf{A}}_k^\top + \partial_{\theta_i} \mathbf{R}\right]\right) \\ &= \text{tr}\left(\mathbf{R}^{-1}\widetilde{\mathbf{A}}_k \partial_{\theta_i} \mathbf{Q} \widetilde{\mathbf{A}}_k^\top\right) + \text{tr}\left(\mathbf{R}^{-1}\partial_{\theta_i} \mathbf{R}\right) \\ &\quad - \text{tr}\left(\mathbf{R}^{-1}\mathbf{U}_{k+1}\mathbf{B}_k(\mathbf{I}_k + \mathbf{T}_k)^{-1}\mathbf{B}_k^\top\mathbf{U}_{k+1}^\top\mathbf{R}^{-1}\widetilde{\mathbf{A}}_k \partial_{\theta_i} \mathbf{Q} \widetilde{\mathbf{A}}_k^\top\right) \\ &\quad - \text{tr}\left(\mathbf{R}^{-1}\mathbf{U}_{k+1}\mathbf{B}_k(\mathbf{I}_k + \mathbf{T}_k)^{-1}\mathbf{B}_k^\top\mathbf{U}_{k+1}^\top\mathbf{R}^{-1}\frac{\partial \mathbf{R}(\bar{\theta})}{\partial \theta_i}\right). \end{aligned}$$

We plug in the expansion for $\widetilde{\mathbf{A}}_k = \mathbf{U}_{k+1}\mathbf{B}_k\mathbf{V}_k^\top$, use the cyclic property of the trace, and identify the matrices Ψ_i^Q and Ψ_i^R to get

$$\begin{aligned} \langle \tilde{\mathbf{Z}}_k^{-1}, \widetilde{\partial_{\theta_i} \mathbf{Z}_k} \rangle_F &= \text{tr}\left(\mathbf{T}_k \Psi_i^Q\right) + \text{tr}\left(\mathbf{R}^{-1}\partial_{\theta_i} \mathbf{R}\right) \\ &\quad - \text{tr}\left((\mathbf{I}_k + \mathbf{T}_k)^{-1}\mathbf{T}_k \Psi_i^Q \mathbf{T}_k\right) - \text{tr}\left((\mathbf{I}_k + \mathbf{T}_k)^{-1}\mathbf{B}_k^\top \Psi_i^R \mathbf{B}_k\right). \end{aligned}$$

In deriving these expressions we have also used the orthogonality relations $\mathbf{U}_{k+1}^\top \mathbf{R}^{-1} \mathbf{U}_{k+1} = \mathbf{I}_{k+1}$. Finally, we can use $\mathbf{I} - (\mathbf{I}_k + \mathbf{T}_k)^{-1}\mathbf{T}_k = (\mathbf{I}_k + \mathbf{T}_k)^{-1}$ to obtain the simplified expression (20).

Appendix B: Error analysis in the objective function

B.1 Proof of Proposition 3.1

Consider the absolute error

$$\bar{\mathcal{F}} - \widehat{\mathcal{F}}_k = \frac{1}{2}T_1 + \frac{1}{2}T_2, \quad (\text{B1})$$

where $T_1 \equiv |\log\det(\mathbf{Z}) - \log\det(\mathbf{Z}_k)|$ and $T_2 \equiv \|\mathbf{A}\boldsymbol{\mu} - \mathbf{d}\|_{\mathbf{Z}^{-1}}^2 - \|\mathbf{A}\boldsymbol{\mu} - \mathbf{d}\|_{\mathbf{Z}_k^{-1}}^2$. We bound each of these terms separately.

For T_1 , first note

$$T_1 = |\log\det(\mathbf{Z}) - \log\det(\mathbf{Z}_k)| = |\log\det(\mathbf{I}_m + \widehat{\mathbf{A}}\widehat{\mathbf{A}}^\top) - \log\det(\mathbf{I}_m + \widehat{\mathbf{A}}_k\widehat{\mathbf{A}}_k^\top)|.$$

Next notice that $\widehat{\mathbf{A}}\widehat{\mathbf{A}}^\top \succeq \widehat{\mathbf{A}}_k\widehat{\mathbf{A}}_k^\top$; apply both parts of [33, Lemma 9] to get

$$T_1 = \log\det(\mathbf{I}_m + \widehat{\mathbf{A}}\widehat{\mathbf{A}}^\top) - \log\det(\mathbf{I}_m + \widehat{\mathbf{A}}_k\widehat{\mathbf{A}}_k^\top) \leq \log\det(\mathbf{I}_m + \widehat{\mathbf{A}}\widehat{\mathbf{A}}^\top - \widehat{\mathbf{A}}_k\widehat{\mathbf{A}}_k^\top).$$

Plug in the SVD of $\widehat{\mathbf{A}}$ and simplify to obtain $T_1 \leq \sum_{j>k} \log(1 + \sigma_j^2(\widehat{\mathbf{A}}))$.

For T_2 , let $\mathbf{r} = \mathbf{R}^{-1/2}(\mathbf{A}\boldsymbol{\mu} - \mathbf{d})$ so that

$$\begin{aligned} T_2 &= \|\mathbf{A}\boldsymbol{\mu} - \mathbf{d}\|_{\mathbf{Z}^{-1}}^2 - \|\mathbf{A}\boldsymbol{\mu} - \mathbf{d}\|_{\mathbf{Z}_k^{-1}}^2 \\ &= \mathbf{r}^\top ((\mathbf{I}_m + \widehat{\mathbf{A}}\widehat{\mathbf{A}}^\top)^{-1} - (\mathbf{I}_m + \widehat{\mathbf{A}}_k\widehat{\mathbf{A}}_k^\top)^{-1})\mathbf{r}. \end{aligned}$$

Applying Cauchy-Schwartz inequality followed by submultiplicativity, we get

$$T_2 \leq \beta_1^2 \|\mathbf{I}_m + \widehat{\mathbf{A}}\widehat{\mathbf{A}}^\top)^{-1} - (\mathbf{I}_m + \widehat{\mathbf{A}}_k\widehat{\mathbf{A}}_k^\top)^{-1}\|_2.$$

Apply [34, Lemma X.1.4] and simplify to get $T_2 \leq \beta_1^2 \sigma_{k+1}^2(\widehat{\mathbf{A}})/(1 + \sigma_{k+1}^2(\widehat{\mathbf{A}}))$.

The proof is complete by plugging in the bounds for T_1 and T_2 into (B1).

B.2 Proof of Proposition 3.2

As in the proof of Proposition 3.1, write

$$\bar{\mathcal{F}} - \tilde{\mathcal{F}}_k = \frac{1}{2}T_1 + \frac{1}{2}T_2, \quad (\text{B2})$$

where $T_1 \equiv \logdet(\mathbf{Z}) - \logdet(\tilde{\mathbf{Z}}_k)$ and $T_2 \equiv \|\mathbf{A}\boldsymbol{\mu} - \mathbf{d}\|_{\mathbf{Z}^{-1}}^2 - \|\mathbf{A}\boldsymbol{\mu} - \mathbf{d}\|_{\tilde{\mathbf{Z}}_k^{-1}}^2$. We bound each of these terms separately.

Let us first consider T_1 . Set $\widehat{\mathbf{A}} = \mathbf{R}^{-1/2}\mathbf{A}\mathbf{Q}^{1/2}$. By Sylvester's determinant identity,

$$\logdet(\mathbf{Z}) = \logdet(\mathbf{R}) + \logdet(\mathbf{I}_m + \widehat{\mathbf{A}}\widehat{\mathbf{A}}^\top). \quad (\text{B3})$$

We can write $\tilde{\mathbf{A}}_k\mathbf{Q}\tilde{\mathbf{A}}_k^\top$ in terms of the projector $\mathbf{\Pi}_{\mathbf{V}_k}$ as

$$\tilde{\mathbf{A}}_k\mathbf{Q}\tilde{\mathbf{A}}_k^\top = \mathbf{A}\mathbf{P}_\mathbf{Q}^\top\mathbf{Q}(\mathbf{A}\mathbf{P}_\mathbf{Q}^\top)^\top = \mathbf{A}\mathbf{Q}^{1/2}\mathbf{\Pi}_{\mathbf{V}_k}(\mathbf{A}\mathbf{Q}^{1/2})^\top,$$

we can assert that $\tilde{\mathbf{Z}}_k = \mathbf{R}^{1/2}[\widehat{\mathbf{A}}\mathbf{\Pi}_{\mathbf{V}_k}\widehat{\mathbf{A}}^\top + \mathbf{I}_m]\mathbf{R}^{1/2}$. Hence,

$$\logdet(\tilde{\mathbf{Z}}_k) = \logdet(\mathbf{R}) + \logdet(\mathbf{I}_m + \widehat{\mathbf{A}}\mathbf{\Pi}_{\mathbf{V}_k}\widehat{\mathbf{A}}^\top) \quad (\text{B4})$$

Set $\mathbf{M} = \widehat{\mathbf{A}}\widehat{\mathbf{A}}^\top$ and $\mathbf{N} = \widehat{\mathbf{A}}\mathbf{\Pi}_{\mathbf{V}_k}\widehat{\mathbf{A}}^\top$. Since the orthogonal projector $\mathbf{\Pi}_{\mathbf{V}_k} \preceq \mathbf{I}_n$ then $\mathbf{N} \preceq \mathbf{M}$ and, by [33, Lemma 9]

$$\begin{aligned} 0 \leq T_1 &\leq \logdet(\mathbf{I}_m + \mathbf{M} - \mathbf{N}) \\ &\leq \text{tr}(\mathbf{M} - \mathbf{N}) = \text{tr}(\mathbf{H}_\mathbf{Q}) - \text{tr}(\widehat{\mathbf{H}}_\mathbf{Q}^{(k)}) \equiv \xi_k, \end{aligned}$$

where in the second step we have used $\logdet(\mathbf{X}) \leq \text{tr}(\mathbf{X} - \mathbf{I}_m)$ for any positive semidefinite matrix $\mathbf{X} \in \mathcal{M}_m(\mathbb{R})$ and in the last step we have used the cyclic and linearity properties of the trace operator.

Next, consider T_2 . Set $\mathbf{r} = \mathbf{R}^{-1/2}(\mathbf{A}\boldsymbol{\mu} - \mathbf{d})$ and recall that $\|\mathbf{r}\|_2 = \beta_1$ in (10). We can rewrite

$$\|\mathbf{A}\boldsymbol{\mu} - \mathbf{d}\|_{\mathbf{Z}^{-1}}^2 = \left\langle (\mathbf{I}_m + \widehat{\mathbf{A}}\widehat{\mathbf{A}}^\top)^{-1}\mathbf{r}, \mathbf{r} \right\rangle_2 = \left\langle f(\widehat{\mathbf{A}}\widehat{\mathbf{A}}^\top)\mathbf{r}, \mathbf{r} \right\rangle_2,$$

where $f : \mathbb{R}_+ \rightarrow \mathbb{R} : x \mapsto 1 - (1+x)^{-1}$. Then using the Cauchy-Schwartz inequality

$$\begin{aligned} T_2 &= \left\langle f(\widehat{\mathbf{A}}\widehat{\mathbf{A}}^\top) - f(\widehat{\mathbf{A}}\boldsymbol{\Pi}_{\mathbf{V}_k}\widehat{\mathbf{A}}^\top)\mathbf{r}, \mathbf{r} \right\rangle_2 \\ &\leq \left\| f(\widehat{\mathbf{A}}\widehat{\mathbf{A}}^\top) - f(\widehat{\mathbf{A}}\boldsymbol{\Pi}_{\mathbf{V}_k}\widehat{\mathbf{A}}^\top) \right\| \beta_1^2. \end{aligned}$$

Since $f(x)$ is operator monotone (see, for example, [34, Exercise V.1.10 (ii)]) and $f(0) = 0$. Hence, by [34, Theorem X.1.1],

$$\begin{aligned} T_2 &\leq \left\| f(\widehat{\mathbf{A}}\widehat{\mathbf{A}}^\top) - f(\widehat{\mathbf{A}}\boldsymbol{\Pi}_{\mathbf{V}_k}\widehat{\mathbf{A}}^\top) \right\|_2 \beta_1^2 \\ &\leq \beta_1^2 f \left(\left\| \widehat{\mathbf{A}}[\mathbf{I}_n - \boldsymbol{\Pi}_{\mathbf{V}_k}]\widehat{\mathbf{A}}^\top \right\|_2 \right) = \frac{\beta_1^2 \left\| \widehat{\mathbf{A}}[\mathbf{I}_n - \boldsymbol{\Pi}_{\mathbf{V}_k}]\widehat{\mathbf{A}}^\top \right\|_2}{1 + \left\| \widehat{\mathbf{A}}[\mathbf{I}_n - \boldsymbol{\Pi}_{\mathbf{V}_k}]\widehat{\mathbf{A}}^\top \right\|_2}. \end{aligned}$$

Since $\mathbf{I}_n - \boldsymbol{\Pi}_{\mathbf{V}_k}$ is an orthogonal projector, it is idempotent and its 2-norm is at most 1; using these two facts

$$\begin{aligned} \left\| \widehat{\mathbf{A}}[\mathbf{I}_n - \boldsymbol{\Pi}_{\mathbf{V}_k}]\widehat{\mathbf{A}}^\top \right\|_2 &= \left\| [\mathbf{I}_n - \boldsymbol{\Pi}_{\mathbf{V}_k}]\widehat{\mathbf{A}}^\top \widehat{\mathbf{A}}[\mathbf{I}_n - \boldsymbol{\Pi}_{\mathbf{V}_k}] \right\|_2 \\ &\leq \left\| [\mathbf{I}_n - \boldsymbol{\Pi}_{\mathbf{V}_k}]\mathbf{H}_Q \right\|_2 \leq |\text{tr}([\mathbf{I}_n - \boldsymbol{\Pi}_{\mathbf{V}_k}]\mathbf{H}_Q)| = \xi_k. \end{aligned}$$

Hence, $T_2 \leq \frac{\beta_1^2 \xi_k}{1 + \xi_k}$. Combine the two bounds as $|\mathcal{F} - \tilde{\mathcal{F}}_k| = |T_1 + T_2| \leq \xi_k + \frac{\beta_1^2 \xi_k}{1 + \xi_k}$. This completes the proof.

Appendix C: Error analysis of trace estimator

To recap the setup: Let \mathbf{H} and \mathbf{Q} be two symmetric positive semidefinite $n \times n$ matrices, with \mathbf{Q} positive definite. Let $\boldsymbol{\omega}_k \in \mathbb{R}^n$ for $1 \leq k \leq N$ be random vectors with independent entries that have zero mean and $\max_j \|(\boldsymbol{\omega}_k)_j\|_{\psi_2} \leq K_\psi$. Define the trace estimator

$$\tilde{\text{tr}}(\mathbf{HQ}) \equiv \frac{1}{N} \sum_{k=1}^N \boldsymbol{\omega}_k^\top \mathbf{HQ} \boldsymbol{\omega}_k.$$

Proof of Proposition 3.3 By the Hanson-Wright inequality [18, Theorem 6.2.1], for a matrix \mathbf{K} and a vector \mathbf{x} with independent mean zero subgaussian entries with subgaussian norm, at most K_ψ

$$\mathbb{P} \left\{ |\mathbf{x}^\top \mathbf{K} \mathbf{x} - \mathbb{E}(\mathbf{x}^\top \mathbf{K} \mathbf{x})| \geq t \right\} \leq 2 \exp \left(-C_{\text{HW}} N \min \left\{ \frac{t^2}{K_\psi^4 \|\mathbf{K}\|_F^2}, \frac{t}{K_\psi^2 \|\mathbf{K}\|_2} \right\} \right). \quad (\text{C5})$$

Apply this result with $\mathbf{K} = \frac{1}{N} \text{diag}(\mathbf{HQ}, \dots, \mathbf{HQ})$ and $\mathbf{x} = [\mathbf{w}_1^\top \dots \mathbf{w}_N^\top]^\top$. Note that $\mathbf{x}^\top \mathbf{K} \mathbf{x} = \tilde{\text{tr}}(\mathbf{HQ})$ and

$$\mathbb{E}(\mathbf{x}^\top \mathbf{K} \mathbf{x}) = \text{tr}(\mathbf{K}) = \text{tr}(\mathbf{HQ}) = \text{tr}(\mathbf{HQ})$$

by the cyclic property of trace, $\|\mathbf{K}\|_F = \|\mathbf{HQ}\|_F / \sqrt{N}$, and $\|\mathbf{K}\|_2 = \|\mathbf{HQ}\|_2 / N$. Plug these identities into (C5) to obtain (24).

For the second result, the tail bound in (24) can be bounded as

$$2 \exp \left(-C_{\text{HW}} N \min \left\{ \frac{t^2}{K^4 \|\mathbf{K}\|_F^2}, \frac{t}{K_\psi^2 \|\mathbf{K}\|_2} \right\} \right) \leq 2 \exp \left(-C_{\text{HW}} N \frac{t^2}{t \|\mathbf{HQ}\|_2 + K_\psi^2 \|\mathbf{HQ}\|_F^2} \right).$$

where we have used the inequality $\min\{x, y\} \geq xy/(x + y)$ for $x, y \geq 0$. Set $t = \epsilon \text{tr}(\mathbf{HQ})$, the upper bound to δ and solve for N . Finally, note that $\text{tr}(\mathbf{HQ}) = \text{tr}(\mathbf{HQ})$ is nonnegative. \square

Acknowledgements This work was partially supported by the National Science Foundation ATD program under grants DMS-2026841/2341843, 2026830, 2026835, and 1845406. 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.

We acknowledge the computing resources provided by North Carolina State University High-Performance Computing Services Core Facility (RRID:SCR_022168). We also thank Andrew Petersen (aapeters@ncsu.edu) for his assistance.

Declarations

The authors have no relevant financial or non-financial interests to disclose.

References

1. Chung, J., Saibaba, A.K.: Generalized hybrid iterative methods for large-scale Bayesian inverse problems. *SIAM J. Sci. Comput.* **39**(5), 24–46 (2017). <https://doi.org/10.1137/16M1081968>
2. Kitanidis, P.K., VoMvoris, E.G.: A geostatistical approach to the inverse problem in groundwater modeling (steady state) and one-dimensional simulations. *Water Resour. Res.* **19**(3), 677–690 (1983)
3. Kitanidis, P.K.: Quasi-linear geostatistical theory for inverting. *Water Resour. Res.* **31**(10), 2411–2419 (1995)
4. Michalak, A.M., Hirsch, A., Bruhwiler, L., Gurney, K.R., Peters, W., Tans, P.P.: Maximum likelihood estimation of covariance parameters for Bayesian atmospheric trace gas surface flux inversions. *J. Geophys. Res. Atmos.* **110**(D24) (2005)
5. Bardsley, J.M.: Computational Uncertainty Quantification for Inverse Problems. Computational Science & Engineering, vol. 19, p. 133. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA (2018)
6. Brown, R.D., Bardsley, J.M., Cui, T.: Semivariogram methods for modeling Whittle–Matérn priors in Bayesian inverse problems. *Inverse Probl.* **36**(5), 055006 (2020)
7. Zhou, Q., Liu, W., Li, J., Marzouk, Y.M.: An approximate empirical Bayesian method for large-scale linear-Gaussian inverse problems. *Inverse Probl.* **34**(9), 095001–18 (2018). <https://doi.org/10.1088/1361-6420/aac287>
8. Majumder, S., Guan, Y., Reich, B.J., Saibaba, A.K.: Kryging: geostatistical analysis of large-scale datasets using Krylov subspace methods. *Stat. Comput.* **32**(5), 74 (2022)

9. Nowak, W., Tenkleva, S., Cirpka, O.A.: Efficient computation of linearized cross-covariance and auto-covariance matrices of interdependent quantities. *Math. Geol.* **35**(1), 53–66 (2003). <https://doi.org/10.1023/A:1022365112368>
10. Saibaba, A.K., Kitanidis, P.K.: Efficient methods for large-scale linear inversion using a geo-statistical approach. *Water Resour. Res.* **48**(5) (2012). <https://doi.org/10.1029/2011WR011778> <https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2011WR011778>
11. Ambikasaran, S., Li, J.Y., Kitanidis, P.K., Darve, E.: Large-scale stochastic linear inversion using hierarchical matrices. *Comput. Geosci.* **17**(6), 913–927 (2013). <https://doi.org/10.1007/s10596-013-9364-0>
12. Franklin, J.N.: Well-posed stochastic extensions of ill-posed linear problems. *J. Math. Anal. Appl.* **31**, 682–716 (1970). [https://doi.org/10.1016/0022-247X\(70\)90017-X](https://doi.org/10.1016/0022-247X(70)90017-X)
13. Lehtinen, M.S., Päiväranta, L., Somersalo, E.: Linear inverse problems for generalised random variables. *Inverse Probl.* **5**(4), 599–612 (1989)
14. Calvetti, D., Somersalo, E.: *Introduction to Bayesian Scientific Computing. Surveys and Tutorials in the Applied Mathematical Sciences*, vol. 2, p. 202. Springer, New York (2007). Ten lectures on subjective computing
15. Reich, B.J., Ghosh, S.K.: *Bayesian Statistical Methods*. CRC Press, Boca Raton, FL (2019)
16. Saibaba, A.K., Chung, J., Petroske, K.: Efficient Krylov subspace methods for uncertainty quantification in large Bayesian linear inverse problems. *Numer. Linear Algebra Appl.* **27**(5), 2325–28 (2020). <https://doi.org/10.1002/nla.2325>
17. Van Loan, C.F.: Generalizing the singular value decomposition. *SIAM J. Numer. Anal.* **13**(1), 76–83 (1976)
18. Vershynin, R.: *High-dimensional Probability: An Introduction with Applications in Data Science*, vol. 47. Cambridge University Press, Cambridge (2018)
19. Cortinovis, A., Kressner, D.: On randomized trace estimates for indefinite matrices with an application to determinants. *Found. Comput. Math.* 1–29 (2021)
20. Byrd, R.H., Gilbert, J.C., Nocedal, J.: A trust region method based on interior point techniques for nonlinear programming. *Math. Program.* **89**(1, Ser. A), 149–185 (2000). <https://doi.org/10.1007/PL00011391>
21. Byrd, R.H., Hribar, M.E., Nocedal, J.: An interior point algorithm for large-scale nonlinear programming. **9**, 877–900 (1999). <https://doi.org/10.1137/S1052623497325107>. Dedicated to John E. Dennis, Jr., on his 60th birthday
22. Waltz, R.A., Morales, J.L., Nocedal, J., Orban, D.: An interior algorithm for nonlinear optimization that combines line search and trust region steps. *Math. Program.* **107**(3, Ser. A), 391–408 (2006). <https://doi.org/10.1007/s10107-004-0560-5>
23. Engl, H.W., Hanke, M., Neubauer, A.: *Regularization of Inverse Problems*. Mathematics and its Applications, vol. 375, p. 321. Kluwer Academic Publishers Group, Dordrecht (1996)
24. Hansen, P.C.: Regularization Tools version 4.0 for Matlab 7.3. *Numer. Algorithm.* **46**(2), 189–194 (2007). <https://doi.org/10.1007/s11075-007-9136-9>
25. Carasso, A.: Determining surface temperatures from interior observations. *SIAM J. Appl. Math.* **42**(3), 558–574 (1982). <https://doi.org/10.1137/0142040>
26. Eldén, L.: Numerical solution of the sideways heat equation. In: *Inverse Problems in Diffusion Processes* (Lake St. Wolfgang, 1994), pp. 130–150. SIAM, Philadelphia, PA (1995)
27. Brunner, H.: *Collocation Methods for Volterra Integral and Related Functional Differential Equations*. Cambridge Monographs on Applied and Computational Mathematics, vol. 15, p. 597. Cambridge University Press, Cambridge (2004). <https://doi.org/10.1017/CBO9780511543234> <https://doi.org/10.1017/CBO9780511543234> <https://doi.org/10.1017/CBO9780511543234>
28. Gazzola, S., Hansen, P.C., Nagy, J.G.: IR Tools: a MATLAB package of iterative regularization methods and large-scale test problems. *Numer. Algorithm.* **81**(3), 773–811 (2019). <https://doi.org/10.1007/s11075-018-0570-7>
29. Chung, J., Jiang, J., Miller, S.M., Saibaba, A.K.: Hybrid projection methods for solution decomposition in large-scale Bayesian inverse problems. *SIAM J. Sci. Comput.* 97–119 (2023)
30. Miller, S.M., Saibaba, A.K., Trudeau, M.E., Mountain, M.E., Andrews, A.E.: Geostatistical inverse modeling with very large datasets: an example from the orbiting carbon observatory 2 (oco-2) satellite. *Geosci. Model. Dev.* **13**(3), 1771–1785 (2020). <https://doi.org/10.5194/gmd-13-1771-2020>
31. NOAA Global Monitoring Laboratory: CarbonTracker - Lagrange (2024). Accessed 2 Aug 2023

32. Cho, T., Chung, J., Miller, S.M., Saibaba, A.K.: Computationally efficient methods for large-scale atmospheric inverse modeling. *Geosci. Model. Dev.* **15**(14), 5547–5565 (2022). <https://doi.org/10.5194/gmd-15-5547-2022>
33. Alexanderian, A., Saibaba, A.K.: Efficient D-optimal design of experiments for infinite-dimensional Bayesian linear inverse problems. *SIAM J. Sci. Comput.* **40**(5), 2956–2985 (2018)
34. Bhatia, R.: Matrix Analysis. Graduate Texts in Mathematics, vol. 169, p. 347. Springer, New York (1997). <https://doi.org/10.1007/978-1-4612-0653-8>. <https://doi.org/prox.lib.ncsu.edu/10.1007/978-1-4612-0653-8>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.