SIAM/ASA J. UNCERTAINTY QUANTIFICATION
Vol. 6, No. 2, pp. 965–989

Low-Rank Eigenvector Compression of Posterior Covariance Matrices for Linear Gaussian Inverse Problems*

Peter Benner†, Yue Qiu‡, and Martin Stoll§

Abstract. We consider the problem of efficient computations of the covariance matrix of the posterior probability density for linear Gaussian Bayesian inverse problems. When the probability density of the noise and the prior are Gaussian, the solution of such a statistical inverse problem is also Gaussian. Therefore, the underlying solution is characterized by the mean and covariance matrix of the posterior probability density. However, the covariance matrix of the posterior probability density is dense and large. Hence, the computation of such a matrix is impossible for large dimensional parameter spaces as is the case for discretized PDEs. Low-rank approximations to the posterior covariance matrix were recently introduced as promising tools. Nevertheless, for transient problems the resulting approximation suffers from an increased dimensionality. We here exploit the structure of the discretized equations in such a way that spatial and temporal components can be separated and the growing complexity is dramatically reduced. In particular, the storage for an eigenvector low-rank approximation up to now was dominated by the computation and storage complexity of $\mathcal{O}(n_x n_t)$, where n_x is the dimension of the spatial domain and n_t is the dimension of the time domain. We develop a new approach that utilizes a low-rank in time algorithm together with the low-rank Hessian method. We reduce both the computational complexity and storage requirement from $\mathcal{O}(n_x n_t)$ to $\mathcal{O}(n_x + n_t)$. We use numerical experiments to illustrate the advantages of our approach.

Key words. Bayesian inverse problems, PDE-constrained optimization, low-rank methods, space-time methods, preconditioning, matrix equations

AMS subject classifications. 65F15, 65F10, 65F50, 93C20, 62F15

DOI. 10.1137/17M1121342

1. Introduction.

Computational mathematicians dealing with simulations of large-scale discretizations describing physical phenomena have had tremendous success over the last decades. This has enabled scientists from various areas of engineering, chemistry, geophysics, etc., to ask more relevant and complex questions. One area that has seen a dramatic increase

*Received by the editors March 16, 2017; accepted for publication (in revised form) April 30, 2018; published electronically June 28, 2018.

http://www.siam.org/journals/juq/6-2/M112134.html

Funding: This work was supported by the European Regional Development Fund (ERDF/EFRE: ZS/2016/04/78156) within the Research Center "Dynamic Systems: Systems Engineering" (CDS). The work was performed while the third author was at the Max Planck Institute for Dynamics of Complex Technical Systems.

†Computational Methods in Systems and Control Theory Group, Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstr. 1, 39106 Magdeburg, Germany (benner@mpi-magdeburg.mpg.de).

‡Corresponding author. Computational Methods in Systems and Control Theory Group, Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstr. 1, 39106 Magdeburg, Germany (qiu@mpi-magdeburg.mpg.de, y.qiu@gmx.us).

§Numerical Linear Algebra for Dynamical Systems Group, Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstr. 1, 39106 Magdeburg, Germany (stollm@mpi-magdeburg.mpg.de), and Professorship of Scientific Computing, Faculty of Mathematics, Technische Universität Chemnitz, 09107 Chemnitz, Germany, (martin.stoll@mathematik.tu-chemnitz.de).

in the number of published results is the field of statistical inverse problems [38, 22, 8]. In particular, the consideration of partial differential equations (PDEs) as models in statistical inverse problems dramatically increases the problem complexity as a refinement of the model in space and time and results in an exponential increase in the problem degrees of freedom. By this we mean that a discretized problem is typically represented by a spatial system matrix $A \in \mathbb{R}^{n_x,n_x}$, where the number of degrees of freedom n_x is typically $O(\frac{1}{h^d})$ with d being the spatial dimension, and h is the mesh size. It is easily seen that halving the mesh size h means the matrix size will grow by a factor of $2, 4, 8, \ldots$ depending on the spatial dimension. This complexity is further increased when the temporal dimension is incorporated.

While numerical analysis has provided many techniques that allow the efficient handling of such problems, e.g., Krylov methods [29] and multigrid techniques [17], we are faced with an even steeper challenge when uncertainty in the parameters of the model is incorporated. For this we consider the approach of Bayesian inverse problems where the goal is to infer posterior mean or posterior by using the prior knowledge combining a set of measured/observed data. While computing the posterior mean typically corresponds to a problem formulation frequently encountered in PDE-constrained optimization [39, 19, 6], the problem of computing the posterior covariance matrix is much more challenging as this matrix is dense and involves the inverse of high-dimensional discretized PDE problems. In [14, 35] and subsequent works, the authors proposed a low-rank approximation of the posterior covariance matrix. This already very efficient approach is based on a low-rank approximation of a space-time matrix, but the individual columns of this low-rank approximation of such a space-time matrix still suffer from the complexity growth, regarding refinement in space and time. Our approach is built on top of this approach by further approximating the space-time eigenvectors in a low-rank form. For this to be applicable, we require an all-at-once[1] discretization, which is feasible for all linear PDEs. This approach extends results in [37] and typically reduces the complexity from $O(n_x n_t)$ to $O(n_x + n_t)$.

Example 1.1. Consider the matrix $\mathbf{A}$ to be the finite difference discretization of a two-dimensional Laplacian with appropriate boundary conditions. It is well known this matrix can be written as $\mathbf{A} = \mathbf{I}_z \otimes \mathbf{D}_y + \mathbf{D}_z \otimes \mathbf{I}_y \in \mathbb{R}^{n_z n_y, n_z n_y}$ with $\mathbf{I}$ and $\mathbf{D}$ the one-dimensional identity and difference discretizations, respectively.

The solution of an eigenvalue problem with $\mathbf{A}$ from Example 1.1 can utilize the particular structure, i.e., the Kronecker products, and therefore allows a much more dramatic reduction of the computation time and storage cost than if we would only consider $\mathbf{A}$ without such a Kronecker product structure. Our methodology developed here will proceed in the same way and we will rely on structures that are similar to that of $\mathbf{A}$.

The paper is therefore organized as follows. We first derive the basic problem following [14]. This is followed by the presentation of a low-rank technique that we previously introduced for PDE-constrained optimization. We then use this to establish a low-rank eigenvalue method based on the classical Lanczos procedure [25] or Arnoldi procedure [30]. After introducing different choices of covariance matrices, we show that our approach can be theoretically justified. We then illustrate the applicability of our proposed methodology to a diffusion problem and

[1] Discretization in space and time simultaneously.

a convection diffusion problem, and we present numerical results illustrating the performance of the scheme.

2. Bayesian inverse problems. We refer to [22, 38] for excellent introductions to the subject of statistical inverse problems. We follow [14, 7] in the derivation of our model setup and start with the *parameter-to-observable map* $g : \mathbb{R}^n \times \mathbb{R}^k \to \mathbb{R}^m$ defined as

$$Y = g(U, E), \tag{2.1}$$

where U, Y, E are vectors of random variables. Note that here, u, our model parameter to be recovered, is a realization of U, the error e is a realization of E, y is a realization of the observables Y, and y_{obs} contains the observed values. As discussed in [7], even when using the "true" model parameters u, the observables y will differ from the measurements y_{obs} due to measurement noise and the inadequacy of the underlying PDE model.

In a typical application such as the one discussed later, evaluating $g(.)$ requires the solution of a PDE potentially coupled to an observation operator representing a domain of interest.

The Bayes' theorem, which plays a key role in the Bayesian inference, is written as

$$\pi_{\text{post}} := \pi(u|y_{\text{obs}}) \propto \pi_{\text{prior}}(u)\pi(y_{\text{obs}}|u), \tag{2.2}$$

where we used the prior probability density function (PDF) $\pi_{\text{prior}}(x)$, the likelihood function $\pi(y_{\text{obs}}|u)$, and the data y_{obs}. The function $\pi_{\text{post}} : \mathbb{R}^n \to \mathbb{R}$ is the posterior PDF and it is specified by the prior PDF and the likelihood function given the observed data. To arrive at a computable expression, we derive the likelihood under the assumption of additive noise

$$Y = f(U) + E, \tag{2.3}$$

where $f : \mathbb{R}^n \to \mathbb{R}^m$ and E is the additive noise. Following [22] we assume that E and U are mutually independent, which means that the known probability density of E when conditioned on $U = u$ is unchanged. Then it holds that Y conditioned on $U = u$ is distributed like E,

$$\pi(y_{\text{obs}}|u) = \pi_{\text{noise}}(y_{\text{obs}} - f(u)).$$

Therefore, Bayes' theorem can be written as

$$\pi_{\text{post}} \propto \pi(u|y_{\text{obs}}) \propto \pi_{\text{prior}}(u)\pi(y_{\text{obs}}|u) = \pi_{\text{prior}}(u)\pi_{\text{noise}}(y_{\text{obs}} - f(u)). \tag{2.4}$$

Assuming that both PDFs for U and E are Gaussian, we can rewrite the PDFs in the form

$$\begin{aligned} \pi_{\text{prior}}(u) &\propto \exp\left(-\frac{1}{2}\left(u - \bar{u}_{\text{prior}}\right)^T \Gamma_{\text{prior}}^{-1}\left(u - \bar{u}_{\text{prior}}\right)\right), \\ \pi_{\text{noise}}(e) &\propto \exp\left(-\frac{1}{2} e^T \Gamma_{\text{noise}}^{-1} e\right), \end{aligned} \tag{2.5}$$

where $\bar{u}_{\text{prior}} \in \mathbb{R}^n$ is the mean of the model parameter prior PDF. We further have the two covariance matrices $\Gamma_{\text{prior}} \in \mathbb{R}^{n,n}$ for the prior and $\Gamma_{\text{noise}} \in \mathbb{R}^{m,m}$ for the noise. Here we

assume that the variables $\bar{u}_{\text{prior}}$, Γ_{prior}, and Γ_{noise} are all known. The Gaussian assumption allows us to rewrite Bayes' theorem further to get

$$\begin{aligned}\pi_{\text{post}} &\propto \exp\left(-\frac{1}{2}\left(u-\bar{u}_{\text{prior}}\right)^T\Gamma_{\text{prior}}^{-1}\left(u-\bar{u}_{\text{prior}}\right)-\frac{1}{2}e^T\Gamma_{\text{noise}}^{-1}e\right)\\ &= \exp\left(-\frac{1}{2}\left\|u-\bar{u}_{\text{prior}}\right\|^2_{\Gamma_{\text{prior}}^{-1}}-\frac{1}{2}\left\|e\right\|^2_{\Gamma_{\text{noise}}^{-1}}\right).\end{aligned} \tag{2.6}$$

Let us further assume that the parameter-to-observable map $g(U,Y)$ is given as in (2.3) with $f(U) = AU$. The matrix $A \in \mathbb{R}^{m,n}$ represents a linear map from the parameters u to the observables y. We will later see that often this matrix involves the inverse of a discretized representation of a PDE operator. Therefore, it will typically be dense and very large. We arrive now at a restated version of the Bayes' theorem (2.3):

$$\pi_{\text{post}} \propto \exp\left(-\frac{1}{2}\left\|u-\bar{u}_{\text{prior}}\right\|^2_{\Gamma_{\text{prior}}^{-1}}-\frac{1}{2}\left\|y_{\text{obs}}-Au\right\|^2_{\Gamma_{\text{noise}}^{-1}}\right). \tag{2.7}$$

From this relation we can express several relevant statistical quantities. For example, we can compute the *maximum a posterior point* (MAP), which is defined via

$$\bar{u}_{\text{post}} = \text{argmax}_u \pi_{\text{post}}(u), \tag{2.8}$$

and to compute it, one can solve the following optimization problem:

$$\bar{u}_{\text{post}} = \text{argmin}_u \left(\frac{1}{2}\left\|u-\bar{u}_{\text{prior}}\right\|^2_{\Gamma_{\text{prior}}^{-1}}+\frac{1}{2}\left\|y_{\text{obs}}-Au\right\|^2_{\Gamma_{\text{noise}}^{-1}}\right).$$

Note that this problem is a deterministic inverse problem and resembles the structure one finds in PDE-constrained optimization problems [39, 21]. For this, many efficient strategies to solve this problem are known. An infinite-dimensional discussion of the above problem is given in [7, 38] and we only refer to the infinite-dimensional setup when needed. Our goal in this paper will not be the solution of the MAP problem. The goal of devising low-rank methods for this case has recently been established in [37] and the techniques there are likely to be applicable as the only difference is the use of the weighting matrices Γ_{noise} and Γ_{prior}, which are for the classical PDE-constrained optimization problem mass matrices or matrices involving mass matrices. The more challenging question lies in the approximation of the posterior covariance matrix

$$\Gamma_{\text{post}} = \left(A^T\Gamma_{\text{noise}}^{-1}A+\Gamma_{\text{prior}}^{-1}\right)^{-1}. \tag{2.9}$$

The approximation of Γ_{post} is in general very costly and, without further approximation, intractable. The approach presented in [14, 7] computes a low-rank approximation to this matrix using the following relation:

$$\begin{aligned}\Gamma_{\text{post}} &= \left(A^T\Gamma_{\text{noise}}^{-1}A+\Gamma_{\text{prior}}^{-1}\right)^{-1}\\ &= \Gamma_{\text{prior}}^{1/2}\left(\Gamma_{\text{prior}}^{1/2}A^T\Gamma_{\text{noise}}^{-1}A\Gamma_{\text{prior}}^{1/2}+I\right)^{-1}\Gamma_{\text{prior}}^{1/2}.\end{aligned} \tag{2.10}$$

The authors in [14, 7] then compute a low-rank approximation to the so-called prior-preconditioned Hessian of the data misfit $\tilde{\mathcal{H}}_{\text{mis}} \in \mathbb{R}^{n,n}$

$$\tilde{\mathcal{H}}_{\text{mis}} = \Gamma_{\text{prior}}^{1/2} A^T \Gamma_{\text{noise}}^{-1} A \Gamma_{\text{prior}}^{1/2} \tag{2.11}$$

with the approximation

$$\tilde{\mathcal{H}}_{\text{mis}} \approx V \Lambda V^T, \tag{2.12}$$

where V and Λ represent the dominant eigenvectors and eigenvalues, respectively. Using this approximation and the Sherman–Morrison–Woodbury formula [15] one obtains for the prior-preconditioned system

$$\left(\Gamma_{\text{prior}}^{1/2} A^T \Gamma_{\text{noise}}^{-1} A \Gamma_{\text{prior}}^{1/2} + I\right)^{-1} \approx \left(V \Lambda V^T + I\right)^{-1} = I - V \tilde{\Lambda} V^T, \tag{2.13}$$

where $\tilde{\Lambda} = \text{diag}(\frac{\lambda_i}{\lambda_i + 1})$, and λ_i is the ith diagonal entry of Λ. In case $\Gamma_{\text{prior}}^{1/2}$ cannot be computed, a low-rank approximation $A^T \Gamma_{\text{noise}}^{-1} A \approx V \Lambda V^T$ can be used for the computations of the posterior covariance matrix in (2.9), given by

$$\begin{aligned} \left(A^T \Gamma_{\text{noise}}^{-1} A + \Gamma_{\text{prior}}^{-1}\right)^{-1} &\approx \left(V \Lambda V^T + \Gamma_{\text{prior}}\right)^{-1} \\ &= \Gamma_{\text{prior}}^{-1} - \Gamma_{\text{prior}}^{-1} V \left(\Lambda^{-1} + V^T \Gamma_{\text{prior}}^{-1} V\right)^{-1} V^T \Gamma_{\text{prior}}^{-1}, \end{aligned} \tag{2.14}$$

and becomes a feasible alternative if $(\Lambda^{-1} + V^T \Gamma_{\text{prior}}^{-1} V)^{-1}$, typically much smaller than the original matrix, can be evaluated in reasonable time.

The approximation $\tilde{\mathcal{H}}_{\text{mis}} \approx V \Lambda V^T$ is already of low-rank form and very effective in reducing the typically infeasible amount of storage. It is very efficient in reducing the complexity of storing only the matrices Λ and V, where the columns of V are dominant eigenvectors of a certain matrix from the discretization a transient problem, and Λ is a diagonal matrix whose diagonal entries are the dominant eigenvalues. For a transient problem, the eigenvectors are still of very large dimension. Other low-rank techniques have been considered recently, e.g., in [9] the authors assume that parameters-to-observable operator is not known and has to be approximated in a fashion similar to (2.12). Follow-up results are found in [10, 35, 34]. Our main goal of this paper is the derivation of an efficient scheme to additionally approximate the matrix V from the low-rank approximation to the misfit Hessian. For this to be applicable, it is important to be able to utilize the structure of the discretized PDE operator and the resulting Kronecker-product form (cf. Example 1.1). Before introducing our solution approach, we need to introduce an idea that becomes instrumental in realizing this and is motivated by a PDE-constrained optimization problem.

Since our main focus is on further reducing the computational and storage cost of the approximation shown in (2.14), we currently do not consider the optimality of the approximation of the prior-preconditioned data-misfit Hessian shown in (2.14). For the details of optimality, we refer to [35] for further reference.

3. A low-rank technique for PDE-constrained optimization. In order to better understand the stochastic inverse problem, we investigate it in relation to a PDE-constrained optimization problem. We start the derivation of the low-rank in time method by considering an often used model problem in PDE-constrained optimization (see [20, 21, 39]), minimization of

$$\min_{y,u} \ \frac{1}{2}\|y - y_{\text{obs}}\|_{\mathcal{Q}}^2 + \frac{\beta}{2}\|u\|_{\mathcal{P}}^2, \tag{3.1}$$

with $\mathcal{P}$ and $\mathcal{Q}$ space-time cylinders. The constraint linking state y and control u of this problem is given by the heat equation with a distributed control term

$$\begin{aligned} y_t - \nabla^2 y &= u \quad \text{in } \Omega, \\ y &= f \quad \text{on } \partial\Omega. \end{aligned}$$

Here Ω denotes the domain and $\partial\Omega$ corresponds to the boundary of the domain. For a more detailed discussion on the well-posedness, existence of solutions, discretization, etc., we refer the interested reader to [20, 21, 39]. The solution of such an optimization problem is obtained using a Lagrangian approach and considering the first order conditions, which for our problem results in a linear system of the form

$$\underbrace{\begin{bmatrix} D_1 \otimes \tau M_1 & 0 & -\left(I_{n_t} \otimes L + C^T \otimes M\right) \\ 0 & D_2 \otimes \beta\tau M_2 & D_3 \otimes \tau N^T \\ -\left(I_{n_t} \otimes L + C \otimes M\right) & D_3 \otimes \tau N & 0 \end{bmatrix}}_{\mathcal{A}} \begin{bmatrix} \mathbf{y} \\ \mathbf{u} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} D_1 \otimes \tau M_1 \mathbf{y}_{\text{obs}} \\ \mathbf{0} \\ \mathbf{d} \end{bmatrix}, \tag{3.2}$$

where $D_1 = D_2 = D_3 = I_{n_t}$ come from the discretization of the temporal parts of the objective function or the right-hand side of the PDE-constraint (cf. [5, 28, 36]). These matrices do not necessarily coincide as this depends on the chosen discretization of the objective function and also on the possibly different terms included in it. The matrices M_1 and M_2 are mass matrices corresponding to observation and control domain. The matrix N is essentially representing the incorporation of the control into the constraint, i.e., N is a mass matrix in the above example. The matrix C represents the all-at-once discretization of the time-derivative in the PDE and L the discretized Laplacian. Here, the state, control, and adjoint state are represented by the following space-time vectors:

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_{n_t} \end{bmatrix}, \mathbf{u} = \begin{bmatrix} \mathbf{u}_1 \\ \vdots \\ \mathbf{u}_{n_t} \end{bmatrix}, \text{ and } \mathbf{p} = \begin{bmatrix} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_{n_t} \end{bmatrix}.$$

We point out again that the Kronecker product is defined as

$$W \otimes V = \begin{bmatrix} w_{11}V & \dots & w_{1m}V \\ \vdots & \ddots & \vdots \\ w_{n1}V & \dots & w_{nm}V \end{bmatrix}$$

and remind the reader of the definition of the vec () operator via

$$\operatorname{vec}(W) = \begin{bmatrix} w_{11} \\ \vdots \\ w_{n1} \\ \vdots \\ w_{nm} \end{bmatrix}$$

as well as the relation

$$\left(W^T \otimes V\right) \operatorname{vec}(Y) = \operatorname{vec}(VYW).$$

In [37], it was shown that the solution to the PDE-constrained optimization problem can be computed in low-rank form,

$$\begin{aligned} Y &= \left[\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_{n_t}\right] \approx W_Y V_Y^T \text{ with } W_Y \in \mathbb{R}^{n_1,k_1}, V_Y \in \mathbb{R}^{n_t,k_1}, \\ U &= \left[\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_{n_t}\right] \approx W_U V_U^T \text{ with } W_U \in \mathbb{R}^{n_2,k_2}, V_U \in \mathbb{R}^{n_t,k_2}, \\ P &= \left[\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_{n_t}\right] \approx W_P V_P^T \text{ with } W_P \in \mathbb{R}^{n_1,k_3}, V_P \in \mathbb{R}^{n_t,k_3}, \end{aligned} \tag{3.3}$$

where the k_i are small in comparison to the spatial and temporal dimensions. The authors in [37] illustrated that the low-rank structure of a right-hand side is maintained throughout a Krylov subspace iteration and the above described representation. Low-rank techniques for Krylov subspace methods have recently received much attention and we refer the reader to [23, 24, 1] and for tensor structured equations [16, 27, 11, 12].

We obtain a significant storage reduction if we can base our approximation of the solution using such low-rank factors. It is easily seen that due to the low-rank nature of the factors, we have to perform fewer multiplications with the submatrices by also maintaining smaller storage requirements.

There are several similarities to the problem (3.1) and the statistical inverse problem presented earlier. It is clear that with the choice

$$\Gamma_{\text{prior}} = \left(D_2 \otimes \beta\tau M_2\right)^{-1} \text{ and } \Gamma_{\text{noise}} = \left(D_1 \otimes \tau M_1\right)^{-1}, \tag{3.4}$$

the PDE-constrained optimization problem can be interpreted as a statistical inverse problem and the posterior covariance matrix Γ_{post} is given by eliminating both state and adjoint state from the system matrix (3.2) to obtain a reduced Hessian system. Furthermore, it is clear that for many choices of prior and noise covariance, we can utilize the tensor structure to compute low-rank solutions. For this we state the posterior covariance matrix of the PDE optimization problem

$$\begin{aligned} \Gamma_{\text{post}} = \Big[&\left(D_2 \otimes \beta\tau M_2\right) + \left(D_3 \otimes \tau N^T\right)\left(I_{n_t} \otimes L + C^T \otimes M\right)^{-1} \\ &\left(D_1 \otimes \tau M_1\right)\left(I_{n_t} \otimes L + C \otimes M\right)^{-1}\left(D_3 \otimes \tau N\right)\Big]^{-1} \end{aligned} \tag{3.5}$$

and the misfit Hessian

$$\begin{aligned} \tilde{\mathcal{H}}_{\text{mis}} = \Big[&\left(D_2 \otimes \beta\tau M_2\right)^{1/2}\left(D_3 \otimes \tau N^T\right)\left(I_{n_t} \otimes L + C^T \otimes M\right)^{-1} \\ &\left(D_1 \otimes \tau M_1\right)\left(I_{n_t} \otimes L + C \otimes M\right)^{-1}\left(D_3 \otimes \tau N\right)\left(D_2 \otimes \beta\tau M_2\right)^{1/2}\Big]. \end{aligned} \tag{3.6}$$

We keep this example in mind when we now discuss a low-rank technique to approximate the eigenvectors of the posterior covariance matrix in low-rank form. For this we propose a low-rank Krylov subspace method to compute the dominating eigenvectors and eigenvalues.

Before discussing the eigenvalue approximation strategy, we want to comment on the scaling of the PDE-constrained optimization problem in relation to the statistical inverse problem discussed in [14]. The authors there consider

$$\min_{y,u} \frac{\beta_{\text{noise}}}{2} \|y - y_{\text{obs}}\|_{\mathcal{Q}}^2 + \frac{\beta_{\text{prior}}}{2} \|u\|_{\mathcal{P}}^2, \tag{3.7}$$

which in the simple case of full observations and control leads to the following rescaling of (3.4):

$$\Gamma_{\text{prior}} = (D_2 \otimes \beta_{\text{prior}} \tau M_2)^{-1} \text{ and } \Gamma_{\text{noise}} = (D_1 \otimes \tau \beta_{\text{noise}} M_1)^{-1}. \tag{3.8}$$

Assuming that $M \approx h^d I$ and $D_i = I$ we get

$$\Gamma_{\text{prior}}^{-1} = \beta_{\text{prior}} \tau h^d I \text{ and } \Gamma_{\text{noise}}^{-1} = \beta_{\text{noise}} \tau h^d I. \tag{3.9}$$

In PDE-constrained optimization one typically reduces β in (3.1) to allow for a more expensive control that drives the state closer to the desired state. This would mean that in the statistical inverse setting $\beta_{\text{noise}} = 1$ and decreasing the value of β_{prior}, which implies that in (2.9) the role of the prior covariance gets diminished and most contributions are coming from the noise. We have similar settings and observations for stochastic inverse problems in this manuscript. These will be shown by the analysis in section 5 and numerical experiments in section 6. The right choice of β_{prior} and β_{noise} depends on the underlying application and we refer to [8] for a discussion of the roles of β_{prior} and β_{noise} as regularization parameters.

4. Low-rank Lanczos/Arnoldi method. We recall that our goal is to find a low-rank approximation of the eigenvectors of the posterior covariance matrix. The goal is to compute an approximation to $\tilde{\mathcal{H}}_{\text{mis}} \approx V \Lambda V^T$ with $V = [v_1, v_2, \ldots, v_k]$ and k much smaller than the dimension of $\tilde{\mathcal{H}}_{\text{mis}}$. For the PDE-constrained optimization problem $\tilde{\mathcal{H}}_{\text{mis}} \in \mathbb{R}^{n_x n_t, n_x n_t}$.

Our main assumption at this point is that storing each v_j and especially a number of such space-time vectors can pose serious problems. Additionally, in order to perform the matrix vector multiplication with $\tilde{\mathcal{H}}_{\text{mis}}$, a large number of PDE-solutions need to be computed. For this we point out that in order to apply the matrix $\tilde{\mathcal{H}}_{\text{mis}}$ in an Arnoldi procedure, we need to solve the spatial system over the whole time domain. A major advantage of our approach is motivated by the fact that

$$v_j = \text{vec}\,(V_j) \quad \forall j = 1, \ldots, k \text{ with } V_j \in \mathbb{R}^{n_x, n_t},$$

which we assume is well approximated via

$$V_j \approx W_{j,1} W_{j,2}^T \tag{4.1}$$

with $W_{j,1} \in \mathbb{R}^{n_x, r_j}$ $W_{j,2} \in \mathbb{R}^{n_t, r_j}$ with $r_j \ll \min\{n_x, n_t\}$. If the Arnoldi or Lanczos vectors are of this form, then the application of the matrix $\tilde{\mathcal{H}}_{\text{mis}}$ to such vectors requires fewer PDE solves than in the full case.

Note that we need to compute the dominant eigenvectors of the prior-preconditioned data misfit Hessian $\tilde{\mathcal{H}}_{\text{mis}}$ (2.11). Therefore the Lanczos method can be used. At the jth Lanczos iteration, we need to perform $\tilde{\mathcal{H}}_{\text{mis}} v_{j-1}$ using the low-rank approach to get a form like (4.1). Here v_{j-1} is the $(j-1)$th Lanczos vector. Due to the low-rank approximation, the orthogonality of Lanczos vectors is lost. Reorthogonalization should be used to orthogonalize Lanczos vectors. Meanwhile, the symmetric property of $\tilde{\mathcal{H}}_{\text{mis}}$ cannot be preserved for the low-rank form of the matrix-vector product. Therefore, we make use of the more general Arnoldi method to compute the dominant eigenvectors of $\tilde{\mathcal{H}}_{\text{mis}}$. We also observe that when applying the Arnoldi method with the truncation error appropriately chosen we still get real eigenvalues. This will be shown in section 6.

We now briefly recall the Arnoldi method, which is the more general procedure. We refer to [15] for details. We recall that the Arnoldi process for a matrix B can be written as

$$BV_k = V_{k+1} H_{k+1,k},$$

where V_k consists of orthornormal columns and $H_{k+1,k} \in \mathbb{R}^{(k+1)\times k}$ is a Hessenberg matrix. The iterative build-up of the columns of V is captured by the recursion

$$\tilde{v}_{k+1} = Bv_k - \sum_{i=1}^{k} h_{i,k} v_i, \tag{4.2}$$

where $h_{i,k} = v_i^T A v_k$. The vector $\tilde{v}_{k+1}$ is then normalized using the scalar $h_{k+1,k}$. While this is well-known our goal here is to illustrate how this method is amenable to the use within a low-rank framework. For the Arnoldi process considered in this manuscript, $B = \tilde{\mathcal{H}}_{\text{mis}}$ and the application of B to v_k results in a low-rank matrix, i.e.,

$$Bv_k = \text{vec}\left(W_{1,B} W_{2,B}^T\right)$$

with small rank. This is because B is related with the inverse of a PDE operator in space and time, which has shown in [37] that applying such an operator to a low-rank vector again gives a low-rank vector. We can then write the right-hand side of (4.2) as

$$\text{vec}\left(W_{1,B} W_{2,B}^T\right) - \alpha_k \,\text{vec}\left(W_{1,k} W_{2,k}^T\right) - \beta_k \,\text{vec}\left(W_{1,k-1} W_{2,k-1}^T\right) \tag{4.3}$$

and write the last expression as

$$\text{vec}\left(\left[W_{1,B}, -\alpha_k W_{1,k}, -\beta_k W_{k-1,B}\right]\left[W_{2,B}, W_{2,k}, W_{2,k-1}\right]^T\right).$$

The size of the matrix

$$\left[W_{1,B}, -\alpha_k W_{1,k}, -\beta_k W_{k-1,B}\right] \in \mathbb{R}^{n_x, r_B + r_k + r_{k-1}}$$

is increased to $r_B + r_k + r_{k-1}$. Using truncation techniques this can typically be controlled. For example, one could achieve the truncation by utilizing skinny QR factorization [23] or truncated singular value decomposition [37].

We apply the more expensive but stable Arnoldi process in our manuscript, where we orthogonalize with respect to all previous Arnoldi vectors. This full reorthogonalization also demands more storage than in the Lanczos case. This is another advantage of our approach. With full reorthogonalization, the storage costs are increasing for both full and low-rank schemes but in the low-rank framework stay significantly below the full scheme. Here, we give the low-rank Arnoldi method in Algorithm 4.1.

Algorithm 4.1. Low-rank Arnoldi method.

1: **Input:** maximal Arnoldi steps m_a, unit vector v_1, truncation tolerance ε_0
2: **for** $j = 1 : \ m_a$ **do**
3: perform low-rank matrix vector product $w = \tilde{\mathcal{H}}_{\text{mis}} v_j$ up to the truncation tolerance ε_0
4: **for** i=1: j **do**
5: perform low-rank dot product $H_{i,\ j} = w^H v_i$
6: update $w \leftarrow w - H_{i,j} v_i$
7: **end for**
8: $H_{j+1,\ j} = \sqrt{w^H w}$
9: **if** $j \ < \ m_a$ **then**
10: $v_{j+1} = {}^1/_{H_{j+1,\ j}} w$
11: **end if**
12: **end for**
13: **Output:** low-rank Arnoldi vectors v_j, and Hessenberg matrix H

We note that the biggest challenge for the low-rank Arnoldi method is to perform the low-rank matrix vector product in line 3 of Algorithm 4.1 since $\tilde{\mathcal{H}}_{\text{mis}}$ is large and dense. We propose the tensor-train (TT) format in section 6 to perform such computations efficiently. The full orthogonalization procedure in lines 4–7 of Algorithm 4.1 is also performed with the TT format.

We use the standard Arnoldi method for low-rank eigenvector computations in Algorithm 4.1. This is practical for the problems studied in this manuscript since we just need to compute up to a few hundred Arnoldi vectors. Since the computational complexity of full orthogonalization increases with the number of Arnoldi vectors, if more Arnoldi vectors are needed, the restarted Arnoldi method can be implemented with a low-rank version [15].

5. Analysis of the eigenfunctions.

The eigenfunction analysis for the general case presented above is not straightforward. Our goal in this section is to give a theoretical justification for simple cases. We start with the case of a steady state problem involving the discretized two-dimensional Poisson equation. For this we consider the misfit Hessian

$$\tilde{\mathcal{H}}_{\text{mis}} = \Gamma_{\text{prior}}^{1/2} A^T \Gamma_{\text{noise}}^{-1} A \Gamma_{\text{prior}}^{1/2}.$$

Assume for now that $\Gamma_{\text{prior}} = \beta_{\text{prior}} I_d$ and $\Gamma_{\text{noise}} = \beta_{\text{noise}} I_d$ are identity matrices of appropriate dimensions, as chosen in [14]. We here assume that the uncertainty comes from the right-hand side of the discretized steady-state Poisson equation and we also assume that the system states

are fully observable. Then, we are left with

$$\tilde{\mathcal{H}}_{\text{mis}} = \frac{\beta_{\text{prior}}}{\beta_{\text{noise}}} A^T A.$$

Note that we do not assume for the data misfit Hessian to be defined in function space as the identity operator in the infinite-dimensional setting does not give a probability measure in nonseparable Hilbert spaces. The point here is to illustrate that the finite difference Laplacian and the above defined covariance matrices define a discrete misfit Hessian with strong decay properties. We consider $A = K^{-1}$ to be the inverse of the finite difference discretized Laplacian. The matrix K discretizes

$$-\Delta u = -u_{xx} - u_{yy}$$

and is defined as

$$K = -\frac{1}{h^2}\begin{bmatrix} T & I & & \\ I & T & \ddots & \\ & \ddots & \ddots & I \\ & & I & T \end{bmatrix} \text{ with } T = \text{tridiag}(1, -4, 1),$$

where we assume zero Dirichlet boundary conditions and thus the problem to be only defined on the inner nodes. Note that for this problem $A^T = A$. Assuming the domain to be the unit square and the matrix to be of dimension n^2, we have the eigenvalues of K given by

$$\lambda_{i,j} = \lambda_i + \lambda_j, \quad 1 \le i, j \le n,$$

with

$$\lambda_l = \frac{2}{h^2}(1 - \cos(l\pi h)) \quad \forall l = 1, \dots, n$$

and

$$u_{ij}^{p,k} = \sin(p\pi i h)\sin(k\pi j h) \text{ with } \quad 1 \le i, j \le n$$

as the eigenvector $u^{(p,k)}$ with $1 \le p, k \le n$. From this expression it is clear that this vector can be written as

$$u^{(p,k)} = v^p \otimes w^k,$$

which means $u^{(p,k)}$ is already separated into two components with separation rank 1 (cf. [18]). Coming back to the misfit Hessian we can write this as

$$\tilde{\mathcal{H}}_{\text{mis}} = \frac{\beta_{\text{prior}}}{\beta_{\text{noise}}} A^2$$

with eigenfunctions as for the Laplacian and eigenvalues given by

$$\mu_{i,j} = \frac{\beta_{\text{prior}}}{\beta_{\text{noise}}} \lambda_{i,j}^{-2},$$

where the decay of $\lambda_{i,j}^{-2}$ is quite rapid. This justifies the approximation of $\tilde{\mathcal{H}}_{\text{mis}}$ by a small number of eigenfunctions. For this case we have established the following lemma.

Lemma 5.1. *The eigenfunctions of the misfit Hessian*

$$\tilde{\mathcal{H}}_{mis} = \frac{\beta_{\text{prior}}}{\beta_{\text{noise}}} A^2$$

with A the inverse Laplacian with zero Dirichlet conditions defined on the unit square, are separated and given by

$$u_{ij}^{p,k} = \sin(p\pi ih)\sin(k\pi jh)$$

as above and hence are of separation rank one.

It is not so straightforward to establish similar results for more complicated equations. For the space-time PDE-constrained optimization problem discussed earlier, we note that

$$VDV^T \approx \alpha(h,\tau,\beta)\left(I_{n_t}\otimes L + C^T\otimes I\right)^{-1}\left(I_{n_t}\otimes L + C\otimes I\right)^{-1},$$

where we used $M \approx h^d I$ and collected all scalars in $\alpha(h,\tau,\beta)$. Our aim is to establish eigenvalue and eigenvector results for

$$\left(I_{n_t}\otimes L + C\otimes I\right)\left(I_{n_t}\otimes L + C^T\otimes I\right).$$

We note that this fits the well-known relation that the singular values of a matrix $\mathcal{A}\in\mathbb{R}^{m,m}$ are the square roots of the eigenvalues of the matrix $\mathcal{A}^T\mathcal{A}$, which, assuming full rank of $\mathcal{A}$, is a symmetric and positive definite matrix. Now assuming the SVDs

$$C = U_C\Sigma_C V_C^T \text{ and } L = U_L\Sigma_L V_L^T,$$

we obtain

$$\underbrace{\left(I_{n_t}\otimes U_L\Sigma_L V_L^T + V_C\Sigma_C U_C^T\otimes I\right)}_{\mathcal{A}} = \underbrace{\left(U_L\otimes V_C\right)}_{U}\underbrace{\left(I_{n_t}\otimes\Sigma_L + \Sigma_C\otimes I\right)}_{\Sigma}\underbrace{\left(V_L^T\otimes U_C^T\right)}_{V^T}.$$

From this it follows that

$$\mathcal{A}^T\mathcal{A} = V\Sigma U^T U\Sigma V^T = V\Sigma^2 V^T$$

is the eigen-decomposition of $\mathcal{A}^T\mathcal{A}$, which has the same eigenvectors as $\mathcal{A}^{-1}\mathcal{A}^{-T}$. As the eigenvalues of $\mathcal{A}^{-1}\mathcal{A}^{-T}$ quickly decay to zero because of the compactness of the operator, we only need a small number of columns of V. Our aim in this paper is to express each column of V further in a low-rank fashion. For this we note that $e_1^{(n_x n_t)} = e_1^{(n_x)}\otimes e_1^{(n_t)}$ and ignoring super indices we get

$$Ve_1 = \left(V_L^T\otimes U_C^T\right)(e_1\otimes e_1) = v_{1,L}^T\otimes u_{1,C}^T$$

and hence a vector of rank one if the eigenvectors are all real. Complex eigenvectors would further introduce a small rank increase and the consideration of M instead of $h^d I$ can with a simultaneous diagonalization of the pencil (L,M) lead to small eigenvector ranks of the overall system. This justifies our choice of approximating the eigenvectors in low-rank form.

6. Numerical results. In this section, we study the performance of the low-rank Lanczos algorithm presented in section 4. The results presented in this section are based on an implementation of the above described algorithms within MATLAB. We perform the discretization of the PDE-operators within the IFISS [31] framework using $Q1$ finite elements for the heat equation and the streamline upwind Petrov–Galerkin (SUPG) method for the convection diffusion equation. Our experiments are performed for a final time $T = 1$ with varying number of time steps. As the domain Ω we consider the unit square but other domains are of course possible. We specify the boundary conditions for each problem separately. Throughout the results section we fixed the truncation at 10^{-8} for which we observed good results. Additionally, we also performed not listed experiments with a tolerance of 10^{-10} for which we also observed fast convergence. Larger tolerances should be combined with a deeper analysis of the algorithms and a combination with flexible outer solvers. All results are performed on a standard Ubuntu desktop Linux machine with an Intel Core i7-4790 CPU @ 3.60GHz and 8GB of RAM.

The mathematical model we consider in this section is given by the instationary PDE

$$\begin{aligned} \frac{\partial}{\partial t} y + \mathcal{L} y &= 0, && \Omega \times (0,T), \\ y &= u, && \Omega \times \{t = 0\}, \\ y &= 0, && \partial\Omega_D \times (0,T), \\ \nabla y \cdot \mathbf{n} &= 0, && \partial\Omega_N \times (0,T), \end{aligned} \tag{6.1}$$

where $\mathcal{L}$ is a PDE operator, and for the numerical experiments we consider the case of the heat equation $\mathcal{L} = -\Delta$ and the convection-diffusion equation $\mathcal{L} = -\mu\Delta + \overrightarrow{\omega} \cdot \nabla$. For all the numerical tests, the initial concentration u represents the unknown parameter $\mathbf{u}$, and the observation data $\mathbf{y}_{\text{obs}}$ are collected by sensors, which are distributed in part of the domain Ω.

As stated in section 2, the statistical inverse problem with Gaussian noise and prior using a Bayesian formulation is related to a weighted least squares problem. Here we use the same functional as in [14], which is given by the functional

$$\min_u \left(\frac{\beta_{\text{noise}}}{2} \int_0^T \int_\Omega (y - y_{\text{obs}})^2 \, b(x,t) dx dt + \frac{\beta_{\text{prior}}}{2} \int_\Omega u^2 dx \right), \tag{6.2}$$

in which u satisfies the PDE model (6.1), $b(x,t)$ is the observation operator, and u is the uncertainty term, which is the initial condition for this numerical example. Here, we study the sparse observation case, where $b(x,t)$ is defined by

$$b(x,t) = \sum_j \delta_j.$$

Here δ_j is the regional function of sensors, $\delta_j = 1$ at the region of the jth sensor and $\delta_j = 0$ elsewhere.

Discretizing the functional (6.2) in turn gives

$$\min_u \left(\frac{1}{2} (\mathbf{y} - \mathbf{y}_{\text{obs}})^T \mathcal{B}^T \Gamma_{\text{noise}}^{-1} \mathcal{B} (\mathbf{y} - \mathbf{y}_{\text{obs}}) + \frac{1}{2} u^T \Gamma_{\text{prior}}^{-1} u \right), \tag{6.3}$$

where $\mathcal{B}$ is the discretization of $b(x,t)$, the variable $\mathbf{y}$ is the discrete variant of y in (6.1) stacked in time, and it satisfies $\mathcal{K}\mathbf{y} = \mathcal{C}u$. Here $\mathcal{K}$ and $\mathcal{C}$ come from the discretization of the PDE model (6.1).

Discretization of the objective function gives that $\Gamma_{\text{noise}} = {}^1/_{\beta_{\text{noise}}} I_{n_t} \otimes M$, and $\Gamma_{\text{prior}} = {}^1/_{\beta_{\text{prior}}} M$, where M is the mass matrix, and I_{n_t} is an $n_t \times n_t$ identity matrix. Here, n_t is the number of time variables. Since we need to compute $\Gamma_{\text{prior}}^{1/2}$, we use $\Gamma_{\text{prior}} \approx {}^1/_{\beta_{\text{prior}}} h^{-d} I_{n_x}$. Here, h is the mesh size, d is the spatial dimension, I_{n_x} is an $n_x \times n_x$ identity matrix, and n_x is the number of spatial variables. Other settings for the covariance operators/matrices are given in [7], where the authors use a scaled diagonal matrix for $\Gamma_{\text{noise}}^{-1}$ and they choose as the prior $\Gamma_{\text{prior}}^{-1} = \mathcal{A}^2$ for a finite-dimensional parameter space. Here $\mathcal{A} = M^{-1}K$, where K stems from the discretization of the weak formulation of

$$-\alpha\Delta u + \alpha u = s \tag{6.4}$$

for some s and Neumann boundary conditions, and M is the mass matrix from a finite element discretization.

Note that both these choices fit perfectly into our framework. This means the low-rank techniques can still be applied but the performance of the method could be different since the low-rank nature is altered by the different prior and noise covariance matrices. This will be illustrated later in section 6.2.

As analyzed in section 2, the posterior covariance matrix Γ_{post} is given by the inverse of the Hessian of (6.3). Therefore,

$$\Gamma_{\text{post}} = \left(\mathcal{C}^T\mathcal{K}^{-T}\mathcal{B}^T\Gamma_{\text{noise}}^{-1}\mathcal{B}\mathcal{K}^{-1}\mathcal{C} + \Gamma_{\text{prior}}^{-1}\right)^{-1}. \tag{6.5}$$

Note that for different PDE models, $\mathcal{K}$ is also different after discretization. In this section, we use two types of PDE models, i.e., the heat equation and the convection-diffusion equation, to show the performance of our low-rank algorithm for the approximation of Γ_{post}. We argue that our low-rank algorithm also applies to other time-dependent PDE operators. Here we apply our method to symmetric systems and unsymmetric systems.

We also point out that due to the uncertainty being given as the initial condition, the posterior Hessian is only of spatial dimension. Nevertheless, the Arnoldi process applied to this matrix requires the solution of space-time problems and the low-rank form of our approach results in a much reduced number of spatial solves. The complexity reduction is even more pronounced when the uncertainty is part of the system as a space-time variable such as the right-hand side of the PDE.

6.1. Implementation details. According to (6.5), the prior-preconditioned data misfit part after discretization of (6.2) is given by

$$\tilde{\mathcal{H}}_{\text{mis}} = \Gamma_{\text{prior}}^{1/2}\mathcal{C}^T\mathcal{K}^{-T}\mathcal{B}^T\Gamma_{\text{noise}}^{-1}\mathcal{B}\mathcal{K}^{-1}\mathcal{C}\Gamma_{\text{prior}}^{1/2}. \tag{6.6}$$

To apply the Lanczos iteration to (6.6), we need to solve the space-time discretized PDE $\mathcal{K}$ and adjoint PDE $\mathcal{K}^T$. Here we take $\mathcal{K}$ as an example. This asks us to solve a linear system of the following type:

$$(I_{n_t} \otimes L + C \otimes M)\, x = f$$

or

$$(I_{n_t} \otimes L + C \otimes M)\, vec(X) = vec(F). \tag{6.7}$$

Here X and F are matrices/tensors of appropriate sizes. Numerical solutions of (6.7) have been studied intensively from the matrix equation point of view; cf. [32, 2, 4] for an overview.

In this manuscript, we solve (6.7) based on its tensor structure and use the alternating minimal energy (AMEn) approach [12] implemented in the TT toolbox [26] to solve the tensor equation (6.7). At each AMEn iteration, either a left Galerkin projection or a right Galerkin projection is applied to the system (6.7). Therefore, after Galerkin projection, we need to solve a linear system of either the format

$$\left(\hat{I}_n \otimes L + \hat{C} \otimes M\right) x = \tilde{b} \tag{6.8}$$

or

$$\left(I_n \otimes \tilde{L} + C \otimes \tilde{M}\right) \tilde{x} = \hat{b}. \tag{6.9}$$

Here $\hat{I}$, $\hat{C}$, $\tilde{L}$, $\tilde{M}$ are matrices of appropriate dimensions after Galerkin projection (cf. [12] for details).

After Galerkin projection, the size of the system (6.8) is still relatively large, while the size of (6.9) is quite moderate. Therefore, Krylov solvers such as the generalized minimal residual (GMRES) [29] method or the induced dimension reduction (IDR(s)) [33] method can be used to solve (6.8), while a direct method can be used to solve (6.9).

To accelerate the convergence of the Krylov solver, we use the preconditioner

$$P = \mathrm{diag}(\hat{I}_n) \otimes L + \mathrm{diag}(C) \otimes M \tag{6.10}$$

to solve (6.8). Here $\mathrm{diag}(\cdot)$ is an operation that extracts the diagonal entries of a matrix and forms a diagonal matrix. One can use standard techniques such as multigrid methods [40] or incomplete LU factorization [15] to approximate the preconditioner (6.10). Here we use backslash implemented in MATLAB.

We also want to point out that there are many other methods to efficiently solve (6.7), such as the low-rank factored alternating directions implicit (ADI) method (cf. [3]).

6.2. The heat equation. In this part, we use the 2D time-dependent heat equation in a unit square as an example to study the performance of our low-rank algorithm for the heat equation. Discretizing the equation in space using Q_1 finite elements and in time using the implicit Euler method gives us an $n_x \times n_t$ linear system, where n_x is the number of spatial variables while n_t is the number of time steps. First we study spectral properties of the prior-preconditioned data misfit part $\tilde{\mathcal{H}}_{\mathrm{mis}}$ and the posterior covariance matrix. Using a 64×64 grid to discretize the heat equation and set $n_t = 30,\ 60,\ 90$, respectively, we plot the 50 largest eigenvalues of $\tilde{\mathcal{H}}_{\mathrm{mis}}$ in Figure 1. Here $\beta_{\mathrm{noise}} = 10^4 \beta_{\mathrm{prior}}$. Note that often our choices for $\beta_{\mathrm{noise}}, \beta_{\mathrm{prior}}$ are somewhat arbitrary as the purpose of the experiments and our method is to provide a tool that gives a robust performance in various if not all setups. The particular choice of the parameters depends on the particular application, the given data, and so on.

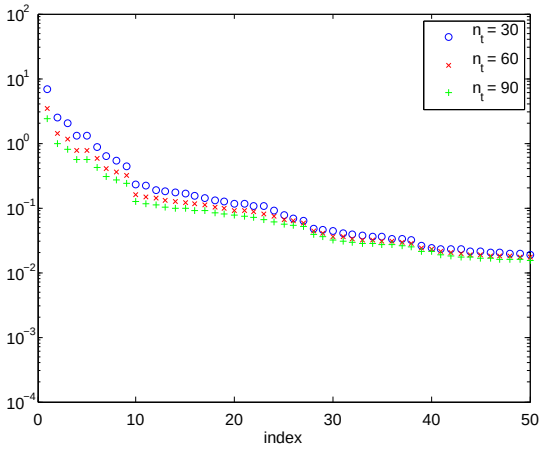


Figure 1. *Largest* 50 *eigenvalues of* $\tilde{\mathcal{H}}_{mis}$ *computed using low-rank Arnoldi.*

As shown in Figure 1, there are only a few dominant eigenvalues of $\tilde{\mathcal{H}}_{\text{mis}}$. For most cases, a threshold of 10^{-1}, or even 10^0, is acceptable to approximate $\tilde{\mathcal{H}}_{\text{mis}}$ and to reduce the uncertainty of the system, which will be shown later. Meanwhile, the number of time steps does not influence the decay rate of $\tilde{\mathcal{H}}_{\text{mis}}$. This makes it possible to compute a fixed number of Arnoldi vectors for even more time steps.

We should point out that the low-rank property $\tilde{\mathcal{H}}_{\text{mis}}$ does not necessarily imply that the eigenvalues of $\tilde{\mathcal{H}}_{\text{mis}}$ should have a significant decay after a few eigenvalue indices and the rest of the eigenvalues are much smaller in magnitude. It can also imply that $\tilde{\mathcal{H}}_{\text{mis}}$ has a few dominant eigenvalues that are much bigger in magnitude than the rest of eigenvalues and the truncation after the first few eigenvalues does give acceptable results, or adding more eigenspaces does not improve the accuracy of the results of interest to a big extent. Similar low-rank properties and results are also observed in [7, 14]. Here we plot the maximum rank used to compute the low-rank approximation of eigenvectors corresponding to the 50 largest eigenvalues for $n_t = 30,\ 60,\ 90$, respectively. In Figure 2, it is shown that the increase of time steps keeps the rank bounded for the low-rank approximation of the eigenvectors of $\tilde{\mathcal{H}}_{\text{mis}}$. The threshold for the low-rank approximation is set to be 10^{-8}.

As illustrated in section 5, the eigenvector also admits a low-rank property. We perform a low-rank approximation on each Arnoldi vector throughout the Arnoldi iteration. Since the low-rank approximation is employed, the orthogonality of the basis of Arnoldi vectors is lost. We just need to compute a few Arnoldi vectors in practice. Here, we use a modified Gram–Schmidt method to perform the full reorthogonalization. Other types of reorthogonalization such as selective orthogonalization [15] or periodic orthogonalization [14] are also possible.

Here we use examples discretized by a 32×32 grid, with 30 and 60 time steps to illustrate the effectiveness of the low-rank Arnoldi method. First, we plot the largest 40 eigenvalues of $\tilde{\mathcal{H}}_{\text{mis}}$ computed using the low-rank Arnoldi method. Next, we compute $\tilde{\mathcal{H}}_{\text{mis}}$ explicitly and use `eigs` in MATLAB to compute its 40 largest eigenvalues. These results are shown in Figure 3. It is clearly shown that the low-rank Arnoldi method can recover the eigenvalues exactly by using reorthogonalization.

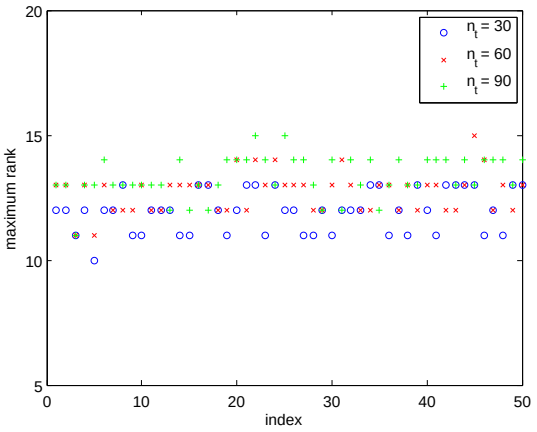
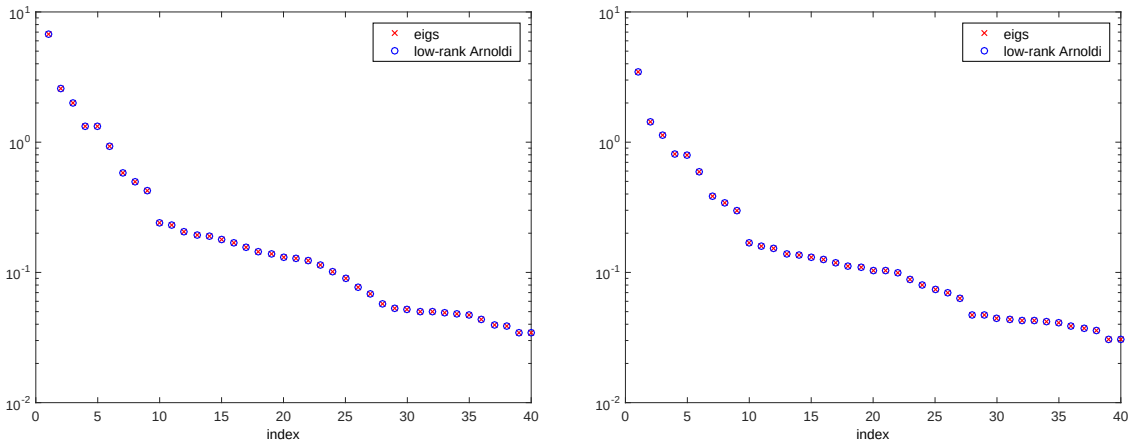


Figure 2. *Maximum rank for eigenvectors of* $\tilde{\mathcal{H}}_{mis}$.



(a) $n_t = 30$ (b) $n_t = 60$

Figure 3. *Eigenvalues of* $\tilde{\mathcal{H}}_{mis}$ *computed using* `eigs` *and low-rank Arnoldi.*

To illustrate that our method can also be applied to other settings for the prior, we choose the prior covariance matrix given by equation (6.4), and we compute the 50 largest eigenvalues of $\tilde{\mathcal{H}}_{\text{mis}}$ for a 32×32 mesh with $n_t = 30$ and $\alpha = 1$ using our low-rank method. We compare with eigenvalues that are explicitly computed using the MATLAB built-in routine `eigs`. We plot these results in Figure 4, which indicates the effectiveness of our method. The plots show that our low-rank approach is applicable to different choices of the prior covariance.

As shown in Figure 1, the increase of time steps does not influence the decay rate of the eigenvectors of $\tilde{\mathcal{H}}_{\text{mis}}$. Next we will show that an increase of the spatial parameters does not influence the decay rate of eigenvalues of $\tilde{\mathcal{H}}_{\text{mis}}$ either. We fix the number of time steps n_t to 30 and compute 60 largest eigenvalues of $\tilde{\mathcal{H}}_{\text{mis}}$; the results are shown in Figure 5(a). The maximum rank used for the low-rank Arnoldi method with different numbers of spatial parameters is shown in Figure 5(b).

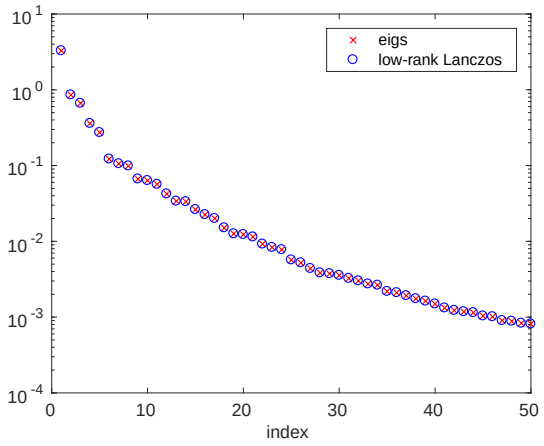
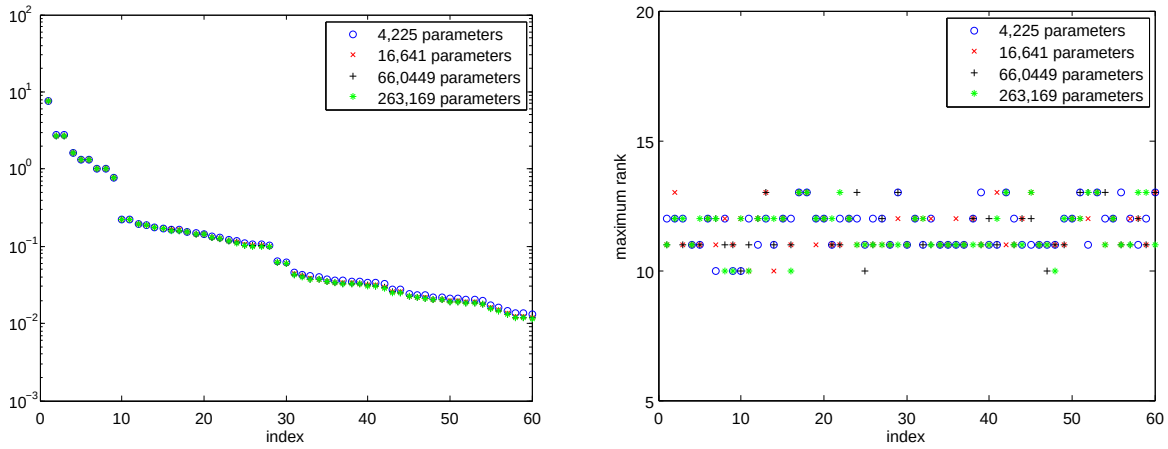


Figure 4. *Largest* 50 *eigenvalues of* $\tilde{\mathcal{H}}_{mis}$ *computed using low-rank Arnoldi and* `eigs`.

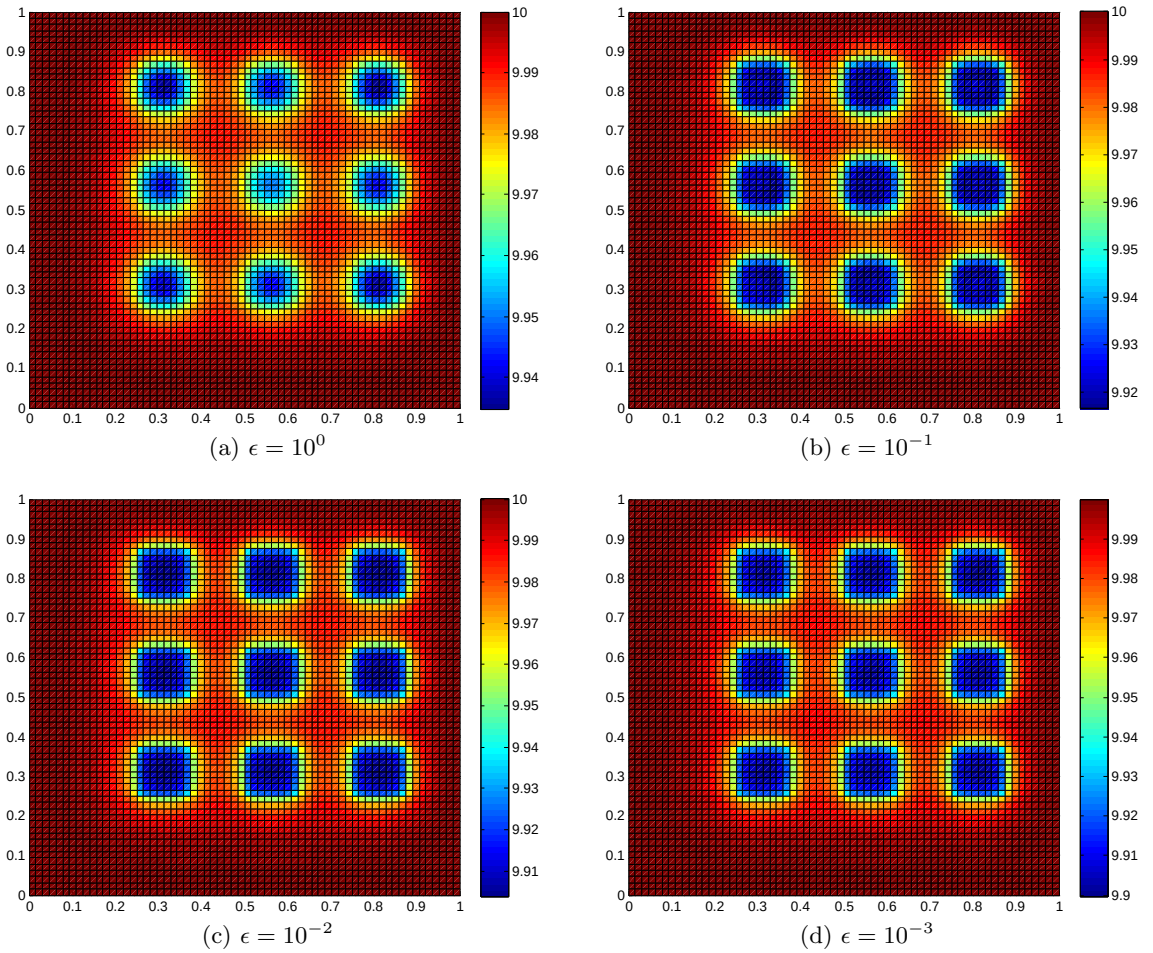


(a) largest 60 eigenvalues of $\tilde{\mathcal{H}}_{\text{mis}}$ (b) maximum rank

Figure 5. *Eigenvalues for* $\tilde{\mathcal{H}}_{mis}$ *and maximum rank for low-rank Arnoldi method,* $n_t = 30$.

Figure 5(a) shows that the increase in number of parameters does not influence the decay property of eigenvalues of $\tilde{\mathcal{H}}_{\text{mis}}$. It is expected that for different number of parameters, the eigenvalues of $\tilde{\mathcal{H}}_{\text{mis}}$ converge to the eigenvalues of the prior-preconditioned operator as long as the discretization of parameter field is good enough. This is illustrated by the eigenvalues shown in Figure 5(a). Meanwhile, maximum ranks used in the low-rank Arnoldi method are also bounded by a constant with the increase of number of parameters, which is shown in Figure 5(b).

As stated before, a threshold of 10^{-1} for the eigenvalue computations of $\tilde{\mathcal{H}}_{\text{mis}}$ is enough to reduce the uncertainty and to approximate the posterior covariance matrix. Next, we plot the diagonal entries of the approximated posterior covariance matrix, i.e., the variance of the points for a 64×64 mesh with a different truncation threshold ϵ for eigenvalue computations of $\tilde{\mathcal{H}}_{\text{mis}}$. We use nine sensors setting for the sparse observation inverse problem, where nine



(a) $\epsilon = 10^0$ (b) $\epsilon = 10^{-1}$

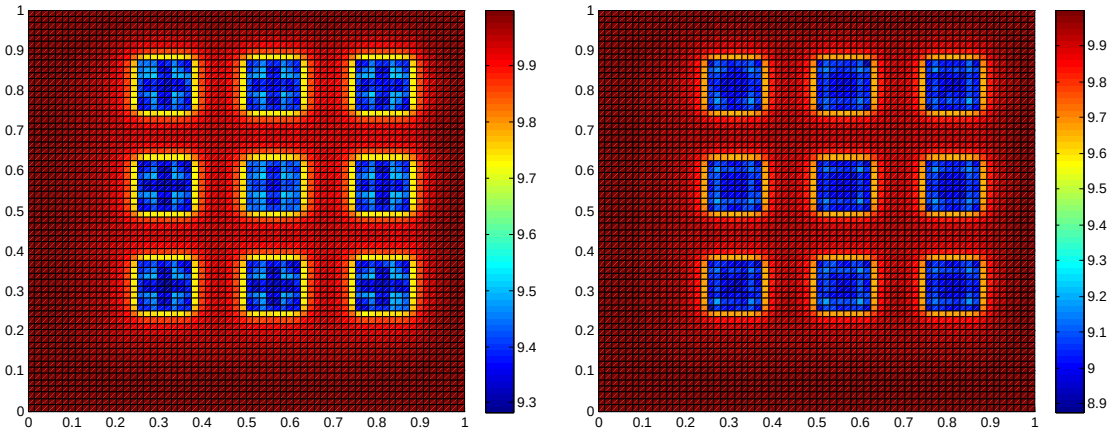
(c) $\epsilon = 10^{-2}$ (d) $\epsilon = 10^{-3}$

Figure 6. *Diagonal entries of* Γ_{post}, $n_t = 30$, $\beta_{noise} = 10^4 \beta_{prior}$.

sensors are uniformly distributed inside the domain and the size of each sensor is $1/256$ of the domain. Here we set $\beta_{\text{noise}} = 10^4 \beta_{\text{prior}}$ and the prior covariance matrix $\Gamma_{\text{prior}} = 10I$, where I is an identity matrix with appropriate size.

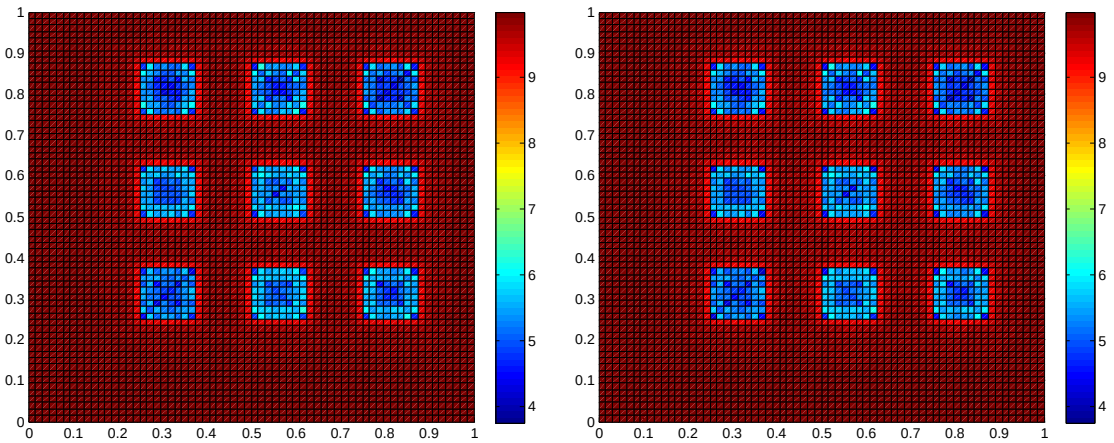
For the sparse observation case, covariance updates are mostly clustered around the area where data are observed, while the rest are dominated by the prior. Uncertainty can only be reduced at areas around the location of sensors. This is clearly shown by Figures 6(a)–6(d), where the dark colored areas are placed at the location of the sensors and have the lowest variance. Decreasing the threshold ϵ, we observe that the variance is further reduced around the location of sensors. For smaller values of ϵ no more reduction in the variance is achieved as all essential information is already captured. Figure 6 shows that as long as the computations of Γ_{post} are convergent, using a threshold $\epsilon = 10^{-1}$ is enough to approximate the posterior covariance matrix and to reduce the uncertainty.

As analyzed in section 5, the eigenvalues of $\tilde{\mathcal{H}}_{\text{mis}}$ are related to $\frac{\beta_{\text{noise}}}{\beta_{\text{prior}}}$ and this ratio gives different updates of the posterior covariance matrix. Next, we use different $\frac{\beta_{\text{noise}}}{\beta_{\text{prior}}}$ ratios to



(a) $\epsilon = 10^0$ (b) $\epsilon = 10^{-1}$

Figure 7. *Diagonal entries of* Γ_{post}*,* $n_t = 30$*,* $\beta_{noise} = 10^6 \beta_{prior}$*.*



(a) $\epsilon = 10^0$ (b) $\epsilon = 10^{-1}$

Figure 8. *Diagonal entries of* Γ_{post}*,* $n_t = 30$*,* $\beta_{noise} = 10^8 \beta_{prior}$*.*

plot the variance of the parameters. These results are shown in Figures 7 and 8. The prior covariance matrix is set to be $\Gamma_{\text{prior}} = 10I$, where I is an identity matrix with appropriate size. For the case $\beta_{\text{noise}} = 10^6 \beta_{\text{prior}}$, we need 72 Arnoldi iterations for $\epsilon = 10^0$, while we need 163 Arnoldi iterations for $\epsilon = 10^{-1}$. With $\beta_{\text{noise}} = 10^8 \beta_{\text{prior}}$, we need 347 Arnoldi iterations for $\epsilon = 10^0$, while we need 470 Arnoldi iterations for $\epsilon = 10^{-1}$.

Figures 7 and 8 show that with the increase of the ratio between β_{noise} and β_{prior}, the diagonal entries of Γ_{post} become smaller. This implies that to further reduce the posterior variance, we need a bigger ratio between β_{noise} and β_{prior}. This can be explained as follows: The weight for the data misfit part in the optimization problem (6.2) is getting bigger for bigger $\frac{\beta_{\text{noise}}}{\beta_{\text{prior}}}$. This means that the data misfit part is more strictly optimized than for smaller $\frac{\beta_{\text{noise}}}{\beta_{\text{prior}}}$. Therefore the error between the estimation and observed data is getting smaller. However,

this yields a more ill-conditioned problem and more Arnoldi iterations are needed. Therefore, a balance between covariance reduction and computational effort is needed with our approach enabling the storage of many Arnoldi vectors due to the complexity reduction of the low-rank approach.

6.3. Convection-diffusion equation. In this section, we study our low-rank approach for a stochastic inverse convection-diffusion problem. Here, the convection-diffusion operator $\mathcal{L}$ is given by

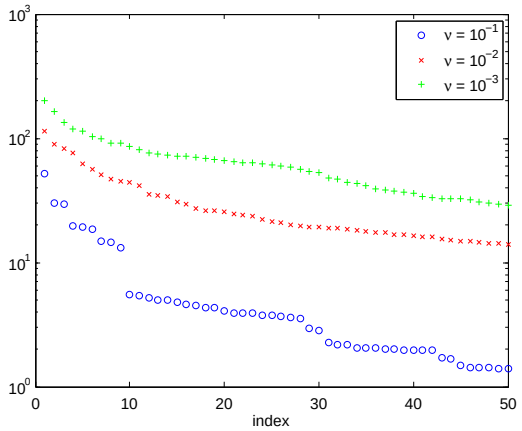
$$\mathcal{L} = -\nu\Delta u + \overrightarrow{\omega} \cdot \nabla u.$$

The computational domain is chosen as a square domain given by $[0,\ 1] \times [0,\ 1]$, $\overrightarrow{\omega} = (0,\ 1)$, and the inflow is posed on the down boundary, while the outflow is posed on the upper boundary. Boundary conditions are prescribed according to the analytic solution of the convection-diffusion equation, which is described as Example 3.3.1 in [13]. We use the SUPG finite element method to discretize the convection-diffusion equation.

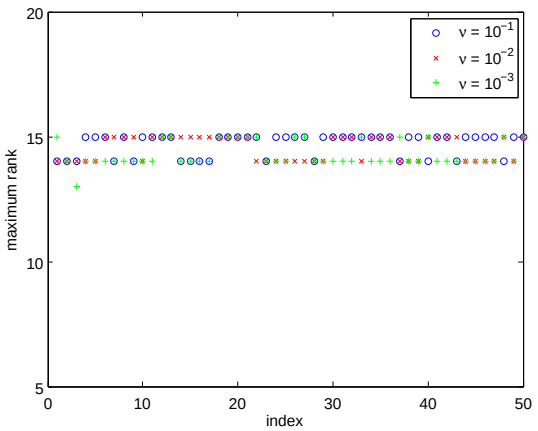
First, we show the eigenvalue decay of $\tilde{\mathcal{H}}_{\text{mis}}$ for different settings of the viscosity parameter ν. Here we set the number of time steps n_t to be 30, and $\beta_{\text{noise}} = 10^4\beta_{\text{prior}}$. We plot the 50 largest eigenvalues of $\tilde{\mathcal{H}}_{\text{mis}}$ for different ν in Figure 9(a).

As shown by Figure 9(a), the eigenvalues of $\tilde{\mathcal{H}}_{\text{mis}}$ decay rapidly for big ν, while this decay rate slows down when ν gets smaller. Therefore, more Arnoldi iterations are needed to get a satisfactory approximation of $\tilde{\mathcal{H}}_{\text{mis}}$. For smaller ν, the largest eigenvalue is also bigger than that for bigger ν, as shown in Figure 9(a). The first few eigenvectors form a more dominant subspace than for bigger ν. It is therefore possible to choose a larger truncation threshold for smaller ν, which will be shown later.

The maximum rank for the low-rank Arnoldi iteration with different ν is shown in Figure 9(b). It shows that the maximum rank does not increase with the decrease of ν and is bounded by a small constant. Therefore, the complexity for both computations and storage is $\mathcal{O}(n_x + n_t)$ for the low-rank Arnoldi approach.



(a) 50 largest eigenvalues of $\tilde{\mathcal{H}}_{\text{mis}}$



(b) maximum ranks at each low-rank Arnoldi iteration

Figure 9. *50 largest eigenvalues of $\tilde{\mathcal{H}}_{mis}$ and maximum ranks at each low-rank Arnoldi iteration with different ν.*

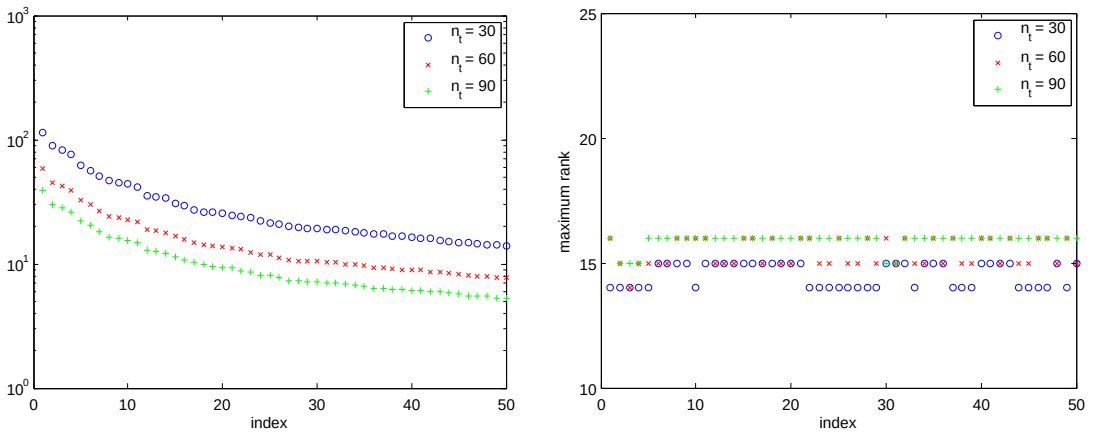
Figure 9(a) shows that the eigenvalues of $\tilde{\mathcal{H}}_{\text{mis}}$ decay slower for smaller ν, and more Arnoldi iterations are needed, which is due to the property of the problem. For such a problem with smaller ν, our low-rank approach is much superior to the standard Arnoldi method introduced in [14] since we need to compute and store more Arnoldi vectors. Note that this is doable with the approach presented here.

Next, we set ν to be 10^{-2} and compute the 50 largest eigenvalues of $\tilde{\mathcal{H}}_{\text{mis}}$ with different n_t, which are shown in Figure 10(a). We are also interested in the relation between the maximum rank at each Arnoldi iteration and the number of time steps n_t, which is shown in Figure 10(b). Figure 10(a) shows that with the increase in the number of time steps, the eigenvalue decay of $\tilde{\mathcal{H}}_{\text{mis}}$ behaves similarly. Maximum ranks at each low-rank Arnoldi iteration with different n_t are bounded by a moderately small constant, which is independent of n_t.

Figures 9(b) and 10(b) show that the maximum rank for each low-rank Arnoldi iteration is almost invariant w.r.t. the number of time steps n_t and the viscosity parameter ν. This makes our low-rank Arnoldi method quite appealing for even complicated stochastic convection dominated inverse problems over a long time horizon.

Next, we show the diagonal entries of Γ_{post} for different settings of ν and the threshold (ϵ) of eigenvalues truncation. Here we set the number of time steps n_t to be 90, use a 32×32 uniform grid to discretize the convection-diffusion equation, and $\beta_{\text{noise}} = 10^4 \beta_{\text{prior}}$. The results are given by Figures 11 and 12.

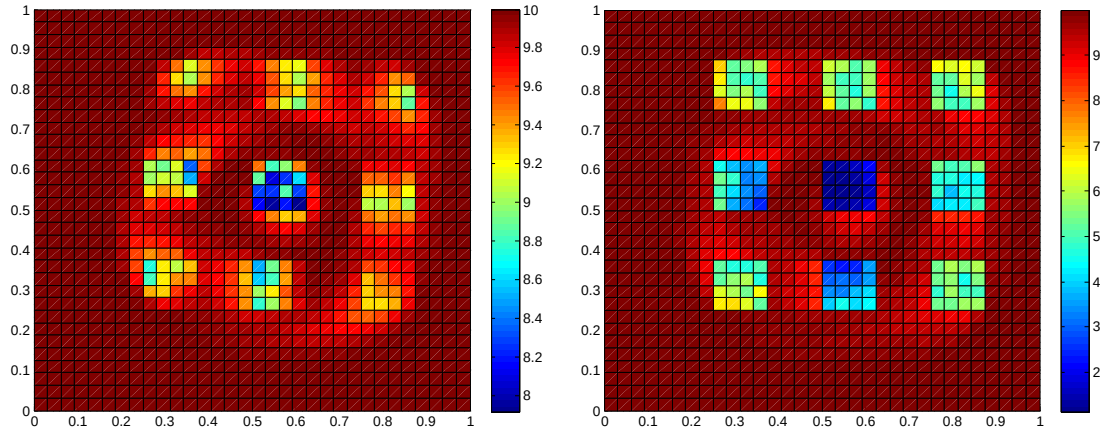
For the case $\nu = 10^{-2}$, we need 17 Arnoldi iterations when we use a threshold $\epsilon = 10^1$, while we need 121 Arnoldi iterations for $\epsilon = 10^0$. For the case $\nu = 10^{-3}$, we need 52 low-rank Arnoldi iterations by setting $\epsilon = 10^1$ and 131 low-rank Lanczos iterations when we use $\epsilon = 10^0$. Note that often a further reduction in the truncation parameter does not yield better results as all the essential information is already captured. Figures 11(b) and 12(b) illustrate that the uncertainty (variance of unknowns) is already reduced dramatically even if we choose a relatively large threshold.



(a) 50 largest eigenvalues of $\tilde{\mathcal{H}}_{\text{mis}}$

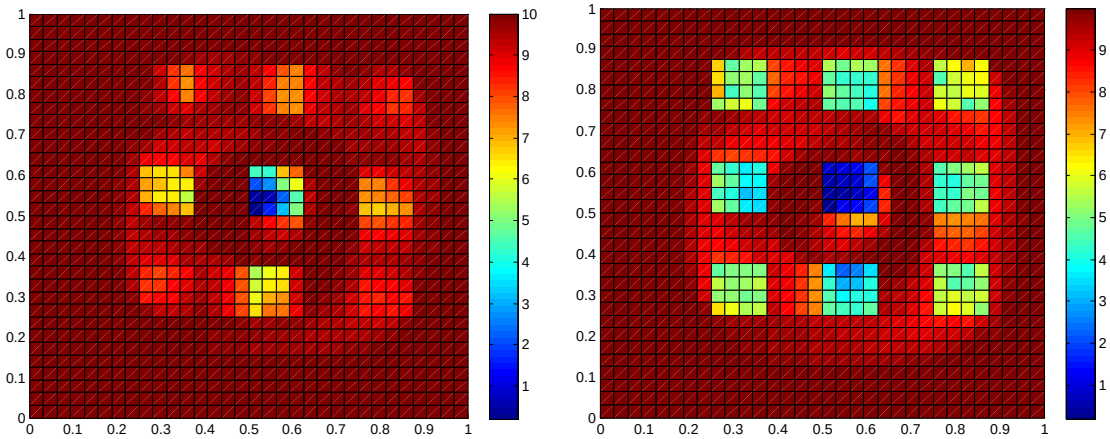
(b) maximum ranks at each low-rank Arnoldi iteration

Figure 10. *50 largest eigenvalues of $\tilde{\mathcal{H}}_{mis}$ and maximum ranks at each low-rank Arnoldi iteration with different n_t.*



(a) $\epsilon = 10^1$ (b) $\epsilon = 10^0$

Figure 11. *Diagonal entries of* Γ_{post}, $\nu = 10^{-2}$.



(a) $\epsilon = 10^1$ (b) $\epsilon = 10^0$

Figure 12. *Diagonal entries of* Γ_{post}, $\nu = 10^{-3}$.

As shown in Figure 9(a), the smaller ν becomes, the bigger the largest eigenvalue of $\tilde{\mathcal{H}}_{\text{mis}}$ is going to be. Therefore, the first few eigenvectors for smaller ν form a more dominant subspace. This in turn implies that the uncertainty is much reduced for smaller ν when we use the same truncation threshold of eigenvalues. We observe this in Figures 11 and 12.

We can conclude that for the stochastic convection-diffusion inverse problem, our low-rank Arnoldi approach is very flexible and efficient for different time horizon lengths and viscosity parameters. It is even preferred for convection dominated stochastic inverse problems with long time horizon.

7. Conclusions.

In this manuscript, we propose a low-rank Arnoldi method to approximate the posterior covariance matrix that appears in stochastic inverse problems. Compared with the standard Arnoldi approach, our approach exploits the low-rank property of each Arnoldi vector and makes a low-rank approximation of such a vector. This reduces the

complexity for both computations and storage demand from $\mathcal{O}(n_x n_t)$ to $\mathcal{O}(n_x + n_t)$. Here n_x is the degree of freedom in space and n_t is the degree of freedom in time. This makes solving large-scale stochastic inverse problems possible.

Our low-rank approach introduced in this manuscript solves linear stochastic inverse problems that can be put into the Bayesian framework. The next step of our work is to extend the low-rank approach introduced in this manuscript to nonlinear stochastic inverse problems, which is still a big challenge.

Acknowledgments. We would like to acknowledge the constructive comments from the two anonymous referees, especially the second referee, for the improvement of this paper.

REFERENCES

[1] R. Andreev and C. Tobler, *Multilevel preconditioning and low-rank tensor iteration for space-time simultaneous discretizations of parabolic PDEs*, Numer. Linear Algebra Appl., 22 (2015), pp. 317–337.

[2] P. Benner, M. Köhler, and J. Saak, *M.E.S.S.—The Matrix Equations Sparse Solvers Library*, http://svncsc.mpi-magdeburg.mpg.de/trac/messtrac.

[3] P. Benner and P. Kürschner, *Computing real low-rank solutions of Sylvester equations by the factored ADI method*, Comput. Math. Appl., 67 (2014), pp. 1656–1672.

[4] P. Benner and J. Saak, *Numerical solution of large and sparse continuous time algebraic matrix Riccati and Lyapunov equations: A state of the art survey*, GAMM-Mitt., 36 (2013), pp. 32–52.

[5] M. Benzi, E. Haber, and L. Taralli, *A preconditioning technique for a class of PDE-constrained optimization problems*, Adv. Comput. Math., 35 (2011), pp. 149–173.

[6] A. Borzí, *Multigrid methods for parabolic distributed optimal control problems*, J. Comput. Appl. Math., 157 (2003), pp. 365–382.

[7] T. Bui-Thanh, O. Ghattas, J. Martin, and G. Stadler, *A computational framework for infinite-dimensional Bayesian inverse problems Part* I: *The linearized case, with application to global seismic inversion*, SIAM J. Sci. Comput., 35 (2013), pp. A2494–A2523.

[8] D. Calvetti and E. Somersalo, *Introduction to Bayesian Scientific Computing*, Surv. Tutor. Appl. Math. Sci., 2, Springer-Verlag, New York, 2007.

[9] J. Chung and M. Chung, *An efficient approach for computing optimal low-rank regularized inverse matrices*, Inverse Problems, 30 (2014), 114009.

[10] J. Chung and M. Chung, *Optimal regularized inverse matrices for inverse problems*, SIAM J. Matrix Anal. Appl., 38 (2017), pp. 458–477.

[11] S. V. Dolgov, *TT-GMRES: Solution to a linear system in the structured tensor format*, Russian J. Numer. Anal. Math. Modelling, 28 (2013), pp. 149–172.

[12] S. V. Dolgov and D. V. Savostyanov, *Alternating minimal energy methods for linear systems in higher dimensions*, SIAM J. Sci. Comput., 36 (2014), pp. A2248–A2271.

[13] H. C. Elman, D. J. Silvester, and A. J. Wathen, *Finite Elements and Fast Iterative Solvers with Applications in Incompressible Fluid Dynamics*, Numer. Math. Sci. Comput., Oxford University Press, New York, 2005.

[14] H. P. Flath, L. C. Wilcox, V. Akçelik, J. Hill, B. van Bloemen Waanders, and O. Ghattas, *Fast algorithms for Bayesian uncertainty quantification in large-scale linear inverse problems based on low-rank partial Hessian approximations*, SIAM J. Sci. Comput., 33 (2011), pp. 407–432.

[15] G. H. Golub and C. F. van Loan, *Matrix Computations*, 3rd ed., Johns Hopkins Stud. Math. Sci., Johns Hopkins University Press, Baltimore, MD, 1996.

[16] L. Grasedyck, D. Kressner, and C. Tobler, *A literature survey of low-rank tensor approximation techniques*, GAMM-Mitt., 36 (2013), pp. 53–78.

[17] W. Hackbusch, *Multigrid Methods and Applications*, Springer Ser. Comput. Math., 4, Springer-Verlag, New York, 1985.

[18] W. Hackbusch, *Tensor Spaces and Numerical Tensor Calculus*, vol. 42 of Springer Ser. Comput. Math., Springer-Verlag, New York, 2012.
[19] R. Herzog and K. Kunisch, *Algorithms for PDE-constrained optimization*, GAMM-Mitt., 33 (2010), pp. 163–176.
[20] M. Hinze, R. Pinnau, M. Ulbrich, and S. Ulbrich, *Optimization with PDE Constraints*, Springer-Verlag, New York, 2009.
[21] K. Ito and K. Kunisch, *Lagrange Multiplier Approach to Variational Problems and Applications*, Adv. Design Control, 15, SIAM, Philadelphia, 2008.
[22] J. Kaipio and E. Somersalo, *Statistical and Computational Inverse Problems*, Appl. Math. Sci., 160, Springer-Verlag, New York, 2005.
[23] D. Kressner and C. Tobler, *Krylov subspace methods for linear systems with tensor product structure*, SIAM J. Matrix Anal. Appl., 31 (2010), pp. 1688–1714.
[24] D. Kressner and C. Tobler, *Low-rank tensor Krylov subspace methods for parametrized linear systems*, SIAM J. Matrix Anal. Appl., 32 (2011), pp. 1288–1316.
[25] C. Lanczos, *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*, J. Res. National Bureau of Standards, 45 (1950), pp. 255–282.
[26] I. V. Oseledets, S. Dolgov, V. Kazeev, D. Savostyanov, O. Lebedeva, P. Zhlobich, T. Mach, and L. Song, *TT-Toolbox*, https://github.com/oseledets/TT-Toolbox.
[27] I. V. Oseledets and S. V. Dolgov, *Solution of linear systems and matrix inversion in the TT-format*, SIAM J. Sci. Comput., 34 (2012), pp. A2718–A2739.
[28] J. W. Pearson, M. Stoll, and A. J. Wathen, *Regularization-robust preconditioners for time-dependent PDE-constrained optimization problems*, SIAM J. Matrix Anal. Appl., 33 (2012), pp. 1126–1152.
[29] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd ed., SIAM Philadelphia, 2003.
[30] Y. Saad, *Numerical Methods for Large Eigenvalue Problems*, Classics in Appl. Math. 66, SIAM, Philadelphia, 2011.
[31] D. Silvester, H. Elman, and A. Ramage, *Incompressible Flow and Iterative Solver Software (IFISS) Version* 3.4, http://www.manchester.ac.uk/ifiss (2015).
[32] V. Simoncini, *Computational methods for linear matrix equations*, SIAM Rev., 58 (2016), pp. 377–441.
[33] P. Sonneveld and M. B. van Gijzen, *IDR(s): A family of simple and fast algorithms for solving large nonsymmetric systems of linear equations*, SIAM J. Sci. Comput., 31 (2008), pp. 1035–1062.
[34] A. Spantini, T. Cui, K. Willcox, L. Tenorio, and Y. Marzouk, *Goal-oriented optimal approximations of Bayesian linear inverse problems*, SIAM J. Sci. Comput., 39 (2017), pp. S167–S196.
[35] A. Spantini, A. Solonen, T. Cui, J. Martin, L. Tenorio, and Y. Marzouk, *Optimal low-rank approximations of Bayesian linear inverse problems*, SIAM J. Sci. Comput., 37 (2015), pp. A2451–A2487.
[36] M. Stoll, *All-at-once solution of a time-dependent time-periodic PDE-constrained optimization problems*, IMA J. Numer. Anal., 34 (2014), pp. 1554–1577.
[37] M. Stoll and T. Breiten, *A low-rank in time approach to PDE-constrained optimization*, SIAM J. Sci. Comput., 37 (2015), pp. B1–B29.
[38] A. M. Stuart, *Inverse problems: A Bayesian perspective*, Acta Numer., 19 (2010), pp. 451–559.
[39] F. Tröltzsch, *Optimal Control of Partial Differential Equations: Theory, Methods and Applications*, AMS, Providence, RI, 2010.
[40] P. Wesseling, *An Introduction to Multigrid Methods*, Pure Appl. Math., John Wiley & Sons, Chichester, UK, 1992.