SIAM/ASA J. Uncertainty Quantification
Vol. 11, No. 1, pp. 235–261

A Fast and Scalable Computational Framework for Large-Scale High-Dimensional Bayesian Optimal Experimental Design[*]

Keyi Wu[†], Peng Chen[‡], and Omar Ghattas[§]

Abstract. We develop a fast and scalable computational framework to solve Bayesian optimal experimental design problems governed by partial differential equations (PDEs) with application to optimal sensor placement by maximizing expected information gain (EIG). Such problems are particularly challenging due to the curse of dimensionality for high-dimensional parameters and the expensive solution of large-scale PDEs. To address these challenges, we exploit two fundamental properties: (1) the low-rank structure of the Jacobian of the parameter-to-observable map, to extract the intrinsically low-dimensional data-informed subspace, and (2) a series of approximations of the EIG that reduce the number of PDE solves while retaining high correlation with the true EIG. Based on these properties, we propose an efficient offline-online decomposition for the optimization problem. The offline stage dominates the cost and entails precomputing all components that require PDE solves. The online stage optimizes sensor placement and does not require any PDE solves. For the online stage, we propose a new greedy algorithm that first places an initial set of sensors using leverage scores and then swaps the selected sensors with other candidates until certain convergence criteria are met, which we call a swapping greedy algorithm. We demonstrate the efficiency and scalability of the proposed method by both linear and nonlinear inverse problems. In particular, we show that the number of required PDE solves is small, independent of the parameter dimension, and only weakly dependent on the data dimension for both problems.

Key words. optimal experimental design, Bayesian inverse problems, expected information gain, swapping greedy algorithm, low-rank approximation, offline-online decomposition

MSC codes. 62K05, 35Q62, 62F15, 35R30, 35Q93, 65C60, 90C27

DOI. 10.1137/21M1466499

1. Introduction.

In many scientific and engineering fields that employ mathematical modeling and computational simulation to predict the behavior of physical systems, uncertainties are ubiquitous. These uncertainties may arise from model coefficients, initial or boundary conditions, external loads, computational geometries, etc. It is crucial to quantify and reduce such uncertainties for more accurate and reliable computational predictions and model-based system optimization. Bayesian inference provides an optimal framework for quantifying the

[*] Received by the editors December 20, 2021; accepted for publication (in revised form) October 21, 2022; published electronically March 3, 2023.
https://doi.org/10.1137/21M1466499
Funding: The work of the authors was partially supported by the DOE ASCR grants DE-SC0019303, DE-SC0021239, the DOD MURI grant FA9550-21-1-0084, and the National Science Foundation grant DMS-2012453.

[†] Department of Mathematics, The University of Texas at Austin, Austin, TX 78705 USA (keyiwu.w@gmail.com).
[‡] School of Computational Science and Engineering, Georgia Institute of Technology, Atlanta, GA 30308 USA (peng@cc.gatech.edu).
[§] Departments of Geological Sciences and Mechanical Engineering and Oden Institute for Computational Engineering and Sciences, The University of Texas at Austin, Austin, TX 78705 USA (omar@oden.utexas.edu).

uncertainties with suitable prior probability distribution based on domain knowledge or expert belief and reducing the uncertainties characterized by their posterior probability distribution through fusing noisy experimental or observational data with the model using Bayes' rule. However, it is challenging to acquire enough data if the experiment is expensive or time-consuming. In this situation, only limited data can be acquired given budget or time constraint. How to design the experiment such that the limited data can reduce the uncertainties as much as possible becomes a very important question and the central task of optimal experimental design (OED) [11, 56, 2, 38].

OED problems can be generally formulated as minimizing or maximizing certain criterion that represents the uncertainty, e.g., the trace (A-optimality) or determinant (D-optimality) of the posterior covariance. In the Bayesian framework, a common choice is the expected information gain (EIG) or mutual information, where the information gain is measured by Kullback–Leibler divergence between the posterior distribution and the prior distribution, and the expected value is taken as an average of this measure over all realizations of the data. Consequently, two integrals are involved in evaluating the EIG, one w.r.t. the posterior distribution and the other w.r.t. the data distribution. Several computational challenges are faced in evaluating the EIG, including (1) evaluation of the double integrals, with one involving integration w.r.t. the posterior distribution, which may require a large number of samples from the posterior distribution, especially if the uncertain parameters are high-dimensional; (2) the parameter-to-observable map at each sample is expensive to evaluate, and thus only a limited number of map evaluations can be afforded, which is often the case when the map evaluation involves the solution of large-scale models, e.g., represented by partial differential equations. For example, in modeling underground fluid flow, one needs to infer the infinite-dimensional permeability field from observations of the fluid velocity or pressure at some locations, or in contaminant diffusion and transportn one needs to infer its source or infinite-dimensional initial concentration field from observations of the concentration at certain locations and time instances. Both of the examples feature large-scale models with high-dimensional parameters after (high-fidelity) discretization. Moreover, when the space of the possible experimental design is also high-dimensional, e.g., the number of candidate sensor locations for the observation is high, one faces the challenge of solving high-dimensional optimization problems, i.e., maximizing the EIG w.r.t. the high-dimensional design variable.

Related work. OED problems with the above computational challenges have attracted increasing attention in recent years. An infinite-dimensional version of the Kullback–Leibler divergence involved in the EIG is studied in [1]. For linear problems (i.e., the parameter-to-observation map is linear in the parameter) the EIG is equivalent to what is called the D-optimal design, which measures the log-determinant of the prior-covariance preconditioned misfit Hessian, whose computation depends on its dominant eigenvalues. Fast decay of the eigenvalues has been proven for some model problems and numerically demonstrated for many others. These include *ice sheet dynamics* [54, 53, 41, 69]; *shape and medium acoustic and electromagnetic scattering* [18, 19, 20, 30, 8, 51, 26]; *seismic wave propagation* [49, 17, 21, 68]; *mantle convection* [63]; *viscous incompressible flow* [66]; *advection-diffusion* [33, 2, 64, 61]; *ocean dynamics* [43]; *turbulent combustion* [28]; *poroelasticity* [37, 6, 7]; *infectious disease spread* [24, 29]; *tumor growth modeling* [60]; *joint inversion* [31]; and *subsurface flow* [3, 4, 27, 23, 25].

Based on this property, efficient methods have been developed to evaluate the D-optimality criterion for infinite-dimensional Bayesian linear problems [57, 58]. In [5], the authors exploited this property and proposed a gradient-based optimization method for D-optimal design, which was extended in [12] for goal-oriented OED. For nonlinear problems, a direct statistical estimator of EIG involves a double-loop Monte Carlo (DLMC) [14], which approximates the EIG via inner and outer Monte Carlo sample average approximations (double loops). Both loops need to generate a large number of samples requiring multiple PDE solves for each sample. Polynomial chaos expansion was employed in [38, 39, 40] as a surrogate for the expensive PDE model, which is, however, not suitable for solving OED problems with high-dimensional parameters. The authors of [46, 47] proposed to approximate the EIG by a Laplace approximation of the posterior distribution, which involves optimization for finding the maximum-a-posterior (MAP) point and eigenvalue decompositions of the prior-preconditioned data misfit Hessian. The optimal design is obtained by an exhaustive search over a prespecified set of experimental scenarios. The authors of [44] considered D-optimal design for quasi-linear problems and used a greedy algorithm to sequentially select observation locations. In [67], the authors used Laplace approximation and investigated two criteria (total flow variance and A-optimal design) with a sparsity-inducing approach and a sum-up rounding approach to find the optimal design. In [48], problems of sensor placement for signal reconstruction with the D-optimality criterion are considered and the accuracy and efficiency between convex optimization and QR pivoting with the greedy method are compared to find the optimal design. However, in all these papers, many expensive PDEs have to be solved in each of the optimization iterations, which may lead to a prohibitively large number of expensive PDEs to solve, especially when the parameter dimension or the data dimension is large. Preliminary work using reduced order models [10, 9] and neural networks [65] has been performed to reduce the computational cost.

Contributions. To address the computational challenges for Bayesian OED problems of maximizing EIG with large-scale models and high-dimensional parameters, we propose a fast and scalable computational framework: fast in that only a limited number of the large-scale models are solved, and scalable in that the computational complexity is independent of both the parameter dimension and the data dimension. These advantages are made possible by (1) using a sequence of approximations of the posterior including Laplace approximation, low-rank approximation of the posterior covariance, and replacement of the design-dependent MAP point by a fixed MAP point, and further by a prior sample point; (2) exploiting an efficient offline-online decomposition of the computation, offline solving all the large-scale models and online solving the model-independent optimization problem; and (3) proposing a swapping greedy algorithm used in the online stage to find the optimal design, with computational complexity dependent only on the dimension of the subspace informed by the data, not on the nominal parameter and data dimensions. More specifically, for linear problems we derive a new approximate form of the D-optimal criterion, whose evaluation at different designs does not involve any PDE solve. Moreover, we propose a swapping greedy algorithm to search for an optimal design. It exploits the dominant data subspace information quantified by the Jacobian of the parameter-to-observable map. For nonlinear problems, we use Laplace approximation and exploit the high correlation of the approximate EIGs with the Laplace approximation centered at different points. Furthermore, we propose an efficient offline-online decomposition of the optimization problem, where the key information is extracted in the offline stage by solving a limited number of PDEs, which is then used in the online stage to find the optimal design by the swapping greedy algorithm. We demonstrate the effectiveness and scalability of our computational framework by two numerical experiments, a linear inverse problem of inferring the initial condition for an advection-diffusion equation and a nonlinear inverse problem of inferring the diffusion coefficient of a log-normal diffusion equation, with both the parameter and data dimensions ranging from a few tens to a few thousands.

Paper overview. In section 2, we review an infinite-dimensional Bayesian inverse problem with finite-dimensional approximation. We also review the EIG optimality criterion, including the specific formulation for sensor placement problems. In section 3, we introduce OED for nonlinear inverse problems with Laplace approximation and its finite-dimensional discretization. We formulate a new framework with several further approximations, employ an online-offline scheme, and introduce a new swapping greedy algorithm to solve the optimization problem. We also discuss the application to linear problems with detailed error analysis. Section 4 presents numerical results for both linear and nonlinear Bayesian inverse problems, followed by the last section 5 for conclusions.

2. Bayesian optimal experimental design.

In this section, we present a general formulation of the Bayesian inverse problem for an abstract forward model and an infinite-dimensional parameter field. The Bayesian inverse problem is discretized by the finite element method, yielding a finite-dimensional problem. We present the optimal experimental design problem as an optimal selection of sensor locations chosen from candidate sensor locations, based on a commonly used design criterion: EIG.

2.1. Bayesian inverse problem.

We consider a PDE model in an abstract form given by

$$\mathcal{R}(u,m)=0 \quad \text{in } \mathcal{V}', \tag{2.1}$$

where u is the state variable defined in a physical domain $\mathcal{D}\subset\mathbb{R}^{n_x}$ with Lipschitz boundary $\partial\mathcal{D}$, where $n_x=1,2,3$, which belongs to a separable Banach space $\mathcal{V}$ with dual $\mathcal{V}'$; m is the parameter field to be inferred, which is assumed to belong to a Hilbert space $\mathcal{M}$ defined in $\mathcal{D}$; and $\mathcal{R}(\cdot,\cdot):\mathcal{V}\times\mathcal{M}\to\mathcal{V}'$ represents the strong form of the PDE, whose weak form can be written as follows: find $u\in\mathcal{V}$ such that

$$r(u,m,v):={}_{\mathcal{V}}\langle v,\mathcal{R}(u,m)\rangle_{\mathcal{V}'}=0 \quad \forall v\in\mathcal{V}, \tag{2.2}$$

where ${}_{\mathcal{V}}\langle\cdot,\cdot\rangle_{\mathcal{V}'}$ denotes the duality pairing.

We assume a data model with additive Gaussian noise, given as

$$\mathbf{y}=\mathcal{B}(u)+\boldsymbol{\varepsilon}, \tag{2.3}$$

where u satisfies the forward problem (2.2) for a given m, and $\mathcal{B}:\mathcal{V}\to\mathbb{R}^{n_y}$ is an observation operator that maps the state variable to the data $\mathbf{y}\in\mathcal{Y}=\mathbb{R}^{n_y}$ at n_y observation points. The observations are corrupted with an additive Gaussian noise $\boldsymbol{\varepsilon}\sim\mathcal{N}(0,\boldsymbol{\Gamma}_{\rm n})$ with symmetric positive definite covariance matrix $\boldsymbol{\Gamma}_{\rm n}\in\mathbb{R}^{n_y\times n_y}$. For notational convenience, we denote the parameter-to-observable map as

$$\mathcal{F}(m)=\mathcal{B}(u(m)). \tag{2.4}$$

We take m to have a Gaussian prior measure $\mu_{\text{pr}} = \mathcal{N}(m_{\text{pr}}, \mathcal{C}_{\text{pr}})$ with mean $m_{\text{pr}} \in \mathcal{M}$ and covariance operator $\mathcal{C}_{\text{pr}} : \mathcal{M} \to \mathcal{M}'$ from $\mathcal{M}$ to its dual $\mathcal{M}'$; $\mathcal{C}_{\text{pr}}$ is a strictly positive self-adjoint operator of trace class. Let $\mathcal{A}$ be a Laplacian-like operator equipped with homogeneous Neumann boundary condition along the boundary ∂D, as in [59, 21]. We take $\mathcal{C}_{\text{pr}} = \mathcal{A}^{-\alpha}$ for sufficiently large $\alpha > 0$, such that $2\alpha > n_x$, to guarantee that $\mathcal{C}_{\text{pr}}$ is of trace class. This choice of prior guarantees a bounded pointwise variance and a well-posed infinite-dimensional Bayesian inverse problem [53].

With the assumption of Gaussian additive noise for $\boldsymbol{\varepsilon}$, the likelihood function $\pi_{\text{like}}(\mathbf{y}|m)$ satisfies

$$\pi_{\text{like}}(\mathbf{y}|m) \propto \exp(-\Phi(m, \mathbf{y})), \tag{2.5}$$

where the potential $\Phi(m, \mathbf{y})$ is given by $\Phi(m, \mathbf{y}) := \frac{1}{2} \|\mathcal{F}(m) - \mathbf{y}\|^2_{\mathbf{\Gamma}_{\text{n}}^{-1}}$, and $||\mathbf{v}||^2_{\mathbf{\Gamma}_{\text{n}}^{-1}} = \mathbf{v}^T \mathbf{\Gamma}_{\text{n}}^{-1} \mathbf{v}$ for any $\mathbf{v} \in \mathbb{R}^{n_y}$. By Bayes' rule, the posterior measure $\mu_{\text{post}}(m|\mathbf{y})$ of the parameter m conditioned on the observational data $\mathbf{y}$ is given by the Radon–Nikodym derivative as

$$\frac{d\mu_{\text{post}}(m|\mathbf{y})}{d\mu_{\text{pr}}(m)} = \frac{1}{Z} \pi_{\text{like}}(\mathbf{y}|m), \tag{2.6}$$

where Z is a normalization constant given by

$$Z = \int_{\mathcal{M}} \pi_{\text{like}}(\mathbf{y}|m) d\mu_{\text{pr}}(m). \tag{2.7}$$

2.2. Expected information gain.

To quantify the information gained from the observational data, different information criteria have been used, e.g., the A-optimal or D-optimal criterion, which use the trace or determinant of the covariance of the posterior [1], respectively, to measure the uncertainty. Here we choose to use the EIG, which is the Kullback–Leibler (KL) divergence between the posterior and the prior, averaged over all data realizations. The KL divergence measures the information gained from data $\mathbf{y}$, which is defined as

$$D_{\text{KL}}(\mu_{\text{post}}(\cdot|\mathbf{y}) \| \mu_{\text{pr}}) := \int_{\mathcal{M}} \ln \left(\frac{d\mu_{\text{post}}(m|\mathbf{y})}{d\mu_{\text{pr}}(m)} \right) d\mu_{\text{post}}(m|\mathbf{y}). \tag{2.8}$$

The EIG, Ψ, takes all possible realizations of the data $\mathbf{y} \in \mathcal{Y}$ into account and is defined as

$$\begin{aligned}
\Psi &:= \mathbb{E}_{\mathbf{y}} \left[D_{\text{KL}}(\mu_{\text{post}}(\cdot|\mathbf{y}) \| \mu_{\text{pr}}) \right] && \text{(2.9a)} \\
&= \int_{\mathcal{Y}} D_{\text{KL}}(\mu_{\text{post}}(\cdot|\mathbf{y}) \| \mu_{\text{pr}}) \, \pi(\mathbf{y}) \, d\mathbf{y} && \text{(2.9b)} \\
&= \int_{\mathcal{Y}} \int_{\mathcal{M}} D_{\text{KL}}(\mu_{\text{post}}(\cdot|\mathbf{y}) \| \mu_{\text{pr}}) \, \pi_{\text{like}}(\mathbf{y}|m) \, d\mu_{\text{pr}}(m) \, d\mathbf{y} && \text{(2.9c)} \\
&= \int_{\mathcal{M}} \int_{\mathcal{Y}} D_{\text{KL}}(\mu_{\text{post}}(\cdot|\mathbf{y}) \| \mu_{\text{pr}}) \, \pi_{\text{like}}(\mathbf{y}|m) \, d\mathbf{y} \, d\mu_{\text{pr}}(m). && \text{(2.9d)}
\end{aligned}$$

In the second equality, $\pi(\mathbf{y})$ is the density of the data $\mathbf{y}$, which follows a Gaussian distribution $\mathcal{N}(\mathcal{F}(m), \mathbf{\Gamma}_{\text{n}})$ conditioned on the parameter m, i.e., $\pi(\mathbf{y}) = \int_{\mathcal{M}} \pi_{\text{like}}(\mathbf{y}|m) \, d\mu_{\text{pr}}(m)$, which is used in the third equality. The fourth equality is obtained by switching the order of integration under the assumption that $D_{\text{KL}}(\mu_{\text{post}}(\cdot|\mathbf{y}) \| \mu_{\text{pr}})$ is integrable by the Fubini theorem. Efficient algorithms for evaluation of the KL divergence and the EIG are presented in section 3.

2.3. Optimal experimental design for sensor placement.

We consider an important subset of OED problems, the sensor placement problem, in which we have a set of candidate sensor locations $\{\mathbf{x}_i\}_{i=1}^d$ in the physical domain $\mathcal{D}$, from which we need to choose r locations (due to a limited budget) to optimally place the sensors for data acquisition. To represent the selected sensor locations, we use a design matrix $\mathbf{W}$, which is a Boolean matrix $\mathbf{W} \in \mathbb{R}^{r\times d}$ such that $\mathbf{W}_{ij} = 1$ if the ith sensor is placed at the jth candidate sensor location, i.e.,

$$\mathbf{W} \in \mathcal{W} := \left\{ \mathbf{W} \in \mathbb{R}^{r\times d} : \mathbf{W}_{ij} \in \{0,1\}, \sum_{j=1}^{d} \mathbf{W}_{ij} = 1, \sum_{i=1}^{r} \mathbf{W}_{ij} \in \{0,1\} \right\}. \tag{2.10}$$

We consider the case of uncorrelated observational noise with covariance matrix

$$\mathbf{\Gamma}_{\mathrm{n}}^d = \mathrm{diag}(\sigma_1^2, \dots, \sigma_d^2) \tag{2.11}$$

corresponding to d candidate sensor locations, where σ_j^2 indicates the noise variance at the jth candidate sensor location. Let $\mathcal{B}_d$ denote the observation operator at all d candidate sensor locations. For a specific design $\mathbf{W}$, we have the observation operator $\mathcal{B} = \mathbf{W}\mathcal{B}_d$, so that the EIG $\Psi(\mathbf{W})$ depends on the design matrix through the observation operator $\mathcal{B}$. To this end, the OED problem can be formulated as follows: find an optimal design matrix $\mathbf{W}^* \in \mathcal{W}$ such that

$$\mathbf{W}^* = \arg\max_{\mathbf{W}\in\mathcal{W}} \Psi(\mathbf{W}). \tag{2.12}$$

2.4. Discretization of the Bayesian inverse problem.

We present a discretization of the Bayesian inverse problem here to facilitate the presentation of our computational algorithms and complexity analysis in the following sections. The parameter field m in the Hilbert space $\mathcal{M}$ is infinite-dimensional. We use a finite element discretization to numerically approximate it in a subspace $\mathcal{M}_n \subset \mathcal{M}$ of dimension n, which is spanned by piecewise continuous Lagrange polynomial basis functions $\{\phi_j\}_{j=1}^n$ defined over a mesh with elements of size h and vertices $\{\mathbf{x}_j\}_{j=1}^n$, such that $\phi_j(\mathbf{x}_i) = \delta_{ij}$, $i,j = 1,\dots,n$, where δ_{ij} denotes the Kronecker delta. The approximation of the parameter $m \in \mathcal{M}$ in $\mathcal{M}_n$, denoted as m_h, can be expressed as

$$m_h = \sum_{j=1}^{n} m_j \phi_j. \tag{2.13}$$

Here we denote $\mathbf{m} = (m_1, \dots, m_n)^T \in \mathbb{R}^n$ as the coefficient vector of m_h.

Let $\mathbf{M}$ denote the finite element mass matrix whose entries are given by

$$\mathbf{M}_{ij} = \int_D \phi_i(\mathbf{x})\phi_j(\mathbf{x})d\mathbf{x}, \; i,j = 1,\dots,n. \tag{2.14}$$

Let $\mathbf{A}$ denote the finite element matrix corresponding to the elliptic differential operator $\mathcal{A}$, i.e.,

$$\mathbf{A}_{ij} = {}_{\mathcal{M}'}\langle \mathcal{A}\phi_j(x), \phi_i(x)\rangle_{\mathcal{M}}, \quad i,j = 1,\dots,n. \tag{2.15}$$

With the specification of the parameter $\alpha = 2$, which satisfies $2\alpha > d$ for $d \le 3$, we obtain a discrete covariance matrix $\mathbf{\Gamma}_{\text{pr}} = \mathcal{A}^{-\alpha}$ corresponding to the covariance operator $\mathcal{C}_{\text{pr}}$, given by

$$\mathbf{\Gamma}_{\text{pr}} = \mathbf{A}^{-1}\mathbf{M}\mathbf{A}^{-1}. \tag{2.16}$$

Then, the prior density for the coefficient vector $\mathbf{m}$, also called the discrete parameter, is given by

$$\pi_{\text{pr}}(\mathbf{m}) \propto \exp\left(-\frac{1}{2}||\mathbf{m} - \mathbf{m}_{\text{pr}}||^2_{\mathbf{\Gamma}_{\text{pr}}^{-1}}\right), \tag{2.17}$$

where $\mathbf{m}_{\text{pr}} \in \mathbb{R}^n$ is the coefficient vector of the approximation of the prior mean m_{pr} in the finite element space $\mathcal{M}_n$. Correspondingly, the posterior density of $\mathbf{m}$ follows the Bayes' rule

$$\pi_{\text{post}}(\mathbf{m}|\mathbf{y}) = \frac{1}{Z}\pi_{\text{like}}(\mathbf{y}|\mathbf{m})\pi_{\text{pr}}(\mathbf{m}), \tag{2.18}$$

where $\pi_{\text{like}}(\mathbf{y}|\mathbf{m})$ is the likelihood function for the discrete parameter $\mathbf{m}$ given by

$$\pi_{\text{like}}(\mathbf{y}|\mathbf{m}) \propto \exp\left(-\Phi(\mathbf{m}, \mathbf{y})\right), \tag{2.19}$$

with the potential

$$\Phi(\mathbf{m}, \mathbf{y}) = \frac{1}{2}||\mathbf{F}(\mathbf{m}) - \mathbf{y}||^2_{\mathbf{\Gamma}_{\text{n}}^{-1}}, \tag{2.20}$$

where $\mathbf{F} : \mathbb{R}^n \to \mathbb{R}^{n_y}$ denotes the discrete parameter-to-observable map corresponding to (2.4). The application of $\mathbf{F}$ involves the solution of the forward model (2.2) by a finite element discretization in a subspace $\mathcal{V}_{n_u} \subset \mathcal{V}$ spanned by basis functions $\{\psi_j\}_{j=1}^{n_u}$, in which the finite element state $u_h = \sum_{j=1}^{n_u} u_j\psi_j$ with coefficient vector $\mathbf{u} = (u_1, \dots, u_{n_u})^T$.

3. Method description.

We consider the Bayesian nonlinear inverse problem for which the parameter-to-observable map $\mathcal{F}(m)$ is nonlinear w.r.t. the parameters. For such problems, we do not have the data-independent EIG as in linear problems, and instead must use a data-averaged KL divergence computed by a sample average approximation as

$$\Psi \approx \frac{1}{N_s}\sum_{i=1}^{N_s} D_{\text{KL}}(\mu_{\text{post}}(\cdot|\mathbf{y}_i)\|\mu_{\text{pr}}), \tag{3.1}$$

where the data $\mathbf{y}_i$ are given by

$$\mathbf{y}_i = \mathcal{F}(m_i) + \boldsymbol{\varepsilon}_i \tag{3.2}$$

with N_s independent and identically distributed (i.i.d.) samples $m_i \sim \mu_{\text{pr}}$ and $\boldsymbol{\varepsilon}_i \sim \mathcal{N}(0, \mathbf{\Gamma}_{\text{n}})$, $i = 1, \dots, N_s$. This involves the significant challenges of (1) computation of the posterior for each data realization, (2) a high-dimensional integral for the KL divergence, and (3) a complex and implicit dependence of the EIG on the design matrix through the solution of the Bayesian nonlinear inverse problem. To tackle these challenges, beyond the low-rank approximation of the Hessian introduced in the last section, we propose a sequence of further approximations including (1) a Laplace approximation of the posterior for each data realization, which leads to an efficient computation of the posterior, (2) an approximation of a varying MAP point for each configuration of the sensors by a fixed MAP point obtained with data from all candidate sensors, thereby avoiding solution of the nonlinear inverse problem for each configuration of the sensors, and (3) an approximation of the fixed MAP point by a synthetic sample drawn from the prior, which further obviates the need to find the MAP point for the nonlinear inverse problem. We demonstrate the accuracy and efficiency of the proposed approximations and derive an optimization problem that can be divided into an "offline" phase that involves expensive solutions of the forward PDE model and an "online" phase that optimize w.r.t. the design matrix and that does not involve PDE solves. We introduce two greedy algorithms for the online phase for scalable and efficient optimization of the design matrix.

3.1. Laplace approximation.

The Laplace approximation replaces the nonlinear Bayesian posterior μ_{post} for given data $\mathbf{y}$ by a Gaussian distribution $\mu_{\text{post}} \approx \mu_{\text{post}}^{\text{LA}}$, with the mean given by the MAP point m_{map} as

$$m_{\text{map}} := \underset{m \in \mathcal{M}}{\operatorname{argmin}} \left(\tfrac{1}{2} \| \mathcal{F}(m) - \mathbf{y} \|^2_{\boldsymbol{\Gamma}_{\text{n}}^{-1}} + \tfrac{1}{2} \| m - m_{\text{pr}} \|^2_{\mathcal{C}_{\text{pr}}^{-1}} \right), \tag{3.3}$$

and the covariance $\mathcal{C}_{\text{post}}$ given by

$$\mathcal{C}_{\text{post}}^{-1} := \mathcal{H}_m(m_{\text{map}}) + \mathcal{C}_{\text{pr}}^{-1}, \tag{3.4}$$

where $\mathcal{H}_m(m_{\text{map}})$ is the Hessian of the data misfit term $\frac{1}{2}\|\mathcal{F}(m) - \mathbf{y}\|^2_{\boldsymbol{\Gamma}_{\text{n}}^{-1}}$ w.r.t. the parameter m, evaluated at the MAP point m_{map}. For efficient computation of the EIG, we employ a Gauss–Newton approximation of the Hessian as (with overloading notation)

$$\mathcal{H}_m(m_{\text{map}}) := \mathcal{J}^*(m_{\text{map}}) \, \boldsymbol{\Gamma}_{\text{n}}^{-1} \, \mathcal{J}(m_{\text{map}}), \tag{3.5}$$

where $\mathcal{J}(m_{\text{map}})$ is the Jacobian of the parameter-to-observable map $\mathcal{F}(m)$ w.r.t. m evaluated at m_{map}, i.e.,

$$\mathcal{J}(m_{\text{map}}) := D_m \mathcal{F}(m)|_{m = m_{\text{map}}}. \tag{3.6}$$

Using the Laplace approximation of the posterior, the analytical form of the KL divergence is given as

$$D_{\text{KL}}(\mu_{\text{post}}^{\text{LA}} \| \mu_{\text{pr}}) = \frac{1}{2} \left[\operatorname{logdet} \left(\mathcal{I} + \widetilde{\mathcal{H}}_m \right) - \operatorname{tr}(\mathcal{C}_{\text{post}}^{\frac{1}{2}} \mathcal{H}_m \mathcal{C}_{\text{post}}^{\frac{1}{2}}) + \| m_{\text{map}} - m_{\text{pr}} \|^2_{\mathcal{C}_{\text{pr}}^{-1}} \right], \tag{3.7}$$

where $\widetilde{\mathcal{H}}_m = \mathcal{C}_{\text{pr}}^{\frac{1}{2}} \, \mathcal{H}_m \, \mathcal{C}_{\text{pr}}^{\frac{1}{2}}$ [1]. Employing a finite-dimensional discretization as in subsection 2.4, we obtain the discrete Laplace approximation with mean $\mathbf{m}_{\text{map}}$ as the coefficient vector of the finite-dimensional basis for m_{map}, and the inverse of the covariance matrix

$$\boldsymbol{\Gamma}_{\text{post}}^{-1} = \mathbf{H}_m(\mathbf{m}_{\text{map}}) + \boldsymbol{\Gamma}_{\text{pr}}^{-1} = \mathbf{J}^T(\mathbf{m}_{\text{map}}) \boldsymbol{\Gamma}_{\text{n}}^{-1} \mathbf{J}(\mathbf{m}_{\text{map}}) + \boldsymbol{\Gamma}_{\text{pr}}^{-1}. \tag{3.8}$$

Here $\mathbf{J}$ is the discretization of the Jacobian $\mathcal{J}$ in (3.6). Algorithms for its efficient computation are deferred to subsection 3.6. Moreover, we have the discrete KL divergence corresponding to (3.7) given by

(3.9)
$$D_{\text{KL}}(\mu_{\text{post}}^{\text{LA}} \| \mu_{\text{pr}}) = \frac{1}{2} \left[\operatorname{logdet} \left(\mathbf{I} + \widetilde{\mathbf{H}}_m(\mathbf{m}_{\text{map}}) \right) - \operatorname{tr}(\boldsymbol{\Gamma}_{\text{post}}^{\frac{1}{2}} \mathbf{H}_m(\mathbf{m}_{\text{map}}) \boldsymbol{\Gamma}_{\text{post}}^{\frac{1}{2}}) + \| \mathbf{m}_{\text{map}} - \mathbf{m}_{\text{pr}} \|^2_{\boldsymbol{\Gamma}_{\text{pr}}^{-1}} \right],$$

where $\widetilde{\mathbf{H}}_m(\mathbf{m}_{\text{map}})$ is the prior-preconditioned (Gauss–Newton) Hessian given by

$$\widetilde{\mathbf{H}}_m(\mathbf{m}_{\text{map}}) = \mathbf{\Gamma}_{\text{pr}}^{\frac{1}{2}}\mathbf{H}_m(\mathbf{m}_{\text{map}})\mathbf{\Gamma}_{\text{pr}}^{\frac{1}{2}}. \tag{3.10}$$

3.2. Low-rank approximation.

Using the Laplace approximation of the Bayesian posterior for given data $\mathbf{y}_i$, $i = 1, \dots, N_s$, we can compute the EIG similar to that for the linear inverse problem by first computing a low-rank approximation of the prior-preconditioned Hessian (3.10). More specifically, we need to compute the eigenvalues of $\widetilde{\mathbf{H}}_m$ at the MAP points $\mathbf{m}^i_{\text{map}}$ for each $i = 1, \dots, N_s$. Letting $\mathbf{J}_d$ denote the Jacobian for all d candidate sensor locations evaluated at $\mathbf{m}^i_{\text{map}}$, we have $\mathbf{J} = \mathbf{W}\mathbf{J}_d$ and $\mathbf{J}^T = \mathbf{J}_d^T\mathbf{W}^T$. Then by introducing $\hat{\mathbf{J}} = \mathbf{\Gamma}_{\text{n}}^{-\frac{1}{2}}\mathbf{J}$, $\hat{\mathbf{J}}_d = (\mathbf{\Gamma}_{\text{n}}^d)^{-\frac{1}{2}}\mathbf{J}_d$ with $\mathbf{\Gamma}_{\text{n}}^d$ defined in (2.11), we have

$$\hat{\mathbf{J}} = \mathbf{\Gamma}_{\text{n}}^{-\frac{1}{2}}\mathbf{J} = \mathbf{\Gamma}_{\text{n}}^{-\frac{1}{2}} W\mathbf{J}_d = W(\mathbf{\Gamma}_{\text{n}}^d)^{-\frac{1}{2}}\mathbf{J}_d = W\hat{\mathbf{J}}_d. \tag{3.11}$$

Hence

$$\widetilde{\mathbf{H}}_m = \mathbf{\Gamma}_{\text{pr}}^{\frac{1}{2}}\mathbf{J}^T\mathbf{\Gamma}_{\text{n}}^{-1}\mathbf{J}\mathbf{\Gamma}_{\text{pr}}^{\frac{1}{2}} = \mathbf{\Gamma}_{\text{pr}}^{\frac{1}{2}}\hat{\mathbf{J}}^T\hat{\mathbf{J}}\mathbf{\Gamma}_{\text{pr}}^{\frac{1}{2}} = \mathbf{\Gamma}_{\text{pr}}^{\frac{1}{2}}\hat{\mathbf{J}}_d^T\mathbf{W}^T\mathbf{W}\hat{\mathbf{J}}_d\mathbf{\Gamma}_{\text{pr}}^{\frac{1}{2}}. \tag{3.12}$$

To compute the eigenvalues of $\widetilde{\mathbf{H}}_m$, we use the following linear algebra fact that, for matrices $\mathbf{A} \in \mathbb{R}^{m\times n}$ and $\mathbf{B} \in \mathbb{R}^{n\times m}$, $\mathbf{AB}$ and $\mathbf{BA}$ have the same nonzero eigenvalues.

Then, the nonzero eigenvalues of $\widetilde{\mathbf{H}}_m = \mathbf{\Gamma}_{\text{pr}}^{\frac{1}{2}}\hat{\mathbf{J}}_d^T\mathbf{W}^T\mathbf{W}\hat{\mathbf{J}}_d\mathbf{\Gamma}_{\text{pr}}^{\frac{1}{2}}$ are the same as the nonzero eigenvalues of $\mathbf{W}\hat{\mathbf{J}}_d\mathbf{\Gamma}_{\text{pr}}\hat{\mathbf{J}}_d^T\mathbf{W}^T := \mathbf{W}\mathbf{H}_d\mathbf{W}^T$. Let

$$\mathbf{H}_d = \hat{\mathbf{J}}_d\mathbf{\Gamma}_{\text{pr}}\hat{\mathbf{J}}_d^T \approx \hat{\mathbf{H}}_d = \mathbf{U}_k\mathbf{\Sigma}_k\mathbf{U}_k^T. \tag{3.13}$$

The low-rank approximation $\hat{\mathbf{H}}_d$ can be efficiently computed by a randomized singular value decomposition (SVD) algorithm. The randomized SVD given in Algorithm 3.1 is a flexible and robust method that requires only Hessian action in a limited number $(2(k+p))$ of given directions instead of forming the full Hessian matrix [35]. This is particularly useful for large-scale problems.

The matrix-free nature of the algorithm is clear in steps 2 and 4, which requires $2(k+p)$ independent data-space Hessian matrix-vector products. Each data-space Hessian matrix-vector product involves a pair of forward and adjoint PDE solves. When the rank k is small and independent of the parameter and data dimensions, as shown in our numerical test, the

Algorithm 3.1 Randomized SVD to compute (3.13).

```
1: Generate i.i.d. Gaussian matrix $\mathbf{\Omega} \in \mathbb{R}^{d\times(k+p)}$ with a small oversampling parameter
   $p$ (e.g., $p = 10$).
2: Compute $\boldsymbol{Y} = \mathbf{H}_d\mathbf{\Omega}$.
3: Compute the QR factorization $\boldsymbol{Y} = \boldsymbol{QR}$ satisfying $\boldsymbol{Q}^T\boldsymbol{Q} = \boldsymbol{I}$.
4: Compute $\boldsymbol{B} = \boldsymbol{Q}^T\mathbf{H}_d\boldsymbol{Q}$.
5: Solve an eigenvalue problem for $\boldsymbol{B}$ such that $\boldsymbol{B} = \boldsymbol{Z}\mathbf{\Sigma}\boldsymbol{Z}^T$.
6: Form $U_k = \boldsymbol{QZ}[1:k]$ and $\Sigma_k = \mathbf{\Sigma}[1:k, 1:k]$.
```

dominant computational cost, i.e., the number of PDE solves, becomes independent of the parameter and data dimensions. Therefore, the eigenvalues of $\widetilde{\mathbf{H}}_m$ can be efficiently computed by first computing the truncated SVD of $\hat{\mathbf{H}}_d$, then forming a small matrix $\mathbf{W}\hat{\mathbf{H}}_d\mathbf{W}^T$ of size $r \times r$, and finally computing the eigenvalues of $\mathbf{W}\hat{\mathbf{H}}_d\mathbf{W}^T$.

Note that both the MAP point $\mathbf{m}^i_{\text{map}}$ and the Hessian $\mathbf{H}_m(\mathbf{m}^i_{\text{map}})$ depend on the design matrix $\mathbf{W}$, with the former depending on $\mathbf{W}$ through the optimization problem (3.3), and the latter through its definition in (3.8), where the Jacobian $\mathbf{J}(\mathbf{m}^i_{\text{map}})$ depends on the design matrix. As a result, the eigenvalues of $\widetilde{\mathbf{H}}_m(\mathbf{m}^i_{\text{map}})$ depend on $\mathbf{W}$. With these eigenvalues $\lambda^i_1, \dots, \lambda^i_r$ of $\mathbf{W}\hat{\mathbf{H}}_d(\mathbf{m}^i_{\text{map}})\mathbf{W}^T$ for each experimental data generated from $\boldsymbol{m}^i$, we obtain the approximations for the quantities in (3.9) as in [61],

(3.14)
$$\text{logdet}\left(\mathbf{I} + \widetilde{\mathbf{H}}_m(\mathbf{m}^i_{\text{map}})\right) \approx \sum_{j=1}^{r} \ln(1 + \lambda^i_j(\mathbf{W})) \text{ and } \text{tr}(\boldsymbol{\Gamma}^{\frac{1}{2}}_{\text{post}}\mathbf{H}_m(\mathbf{m}^i_{\text{map}})\boldsymbol{\Gamma}^{\frac{1}{2}}_{\text{post}}) \approx \sum_{j=1}^{r} \frac{\lambda^i_j(\mathbf{W})}{1 + \lambda^i_j(\mathbf{W})},$$

which leads to a Laplace approximation (with Gauss–Newton Hessian and low-rank approximation) of the EIG (3.1) as

$$\hat{\Psi}(\mathbf{W}) = \frac{1}{N_s}\sum_{i=1}^{N_s}\frac{1}{2}\left[\sum_{j=1}^{r}\left(\ln(1+\lambda^i_j(\mathbf{W})) - \frac{\lambda^i_j(\mathbf{W})}{1+\lambda^i_j(\mathbf{W})}\right) + \|\mathbf{m}^i_{\text{map}}(\mathbf{W}) - \mathbf{m}_{\text{pr}}\|^2_{\boldsymbol{\Gamma}^{-1}_{\text{pr}}}\right]. \tag{3.15}$$

3.3. Fixed MAP point approximation.

To evaluate the Laplace approximation of the EIG in (3.15) for each design matrix $\mathbf{W}$, one has to solve the optimization problem (3.3) for the MAP point $\mathbf{m}^i_{\text{map}}(\mathbf{W})$ for each data $\boldsymbol{y}_i$, $i = 1, \dots, N_s$, which is usually very expensive. Rather than solve the optimization problem for each design $\mathbf{W}$ and each data $\boldsymbol{y}_i$, we consider the MAP point $\mathbf{m}^i_{\text{map}}(\mathbf{W}_{\text{all}})$ at data $\boldsymbol{y}_i$ observed from all candidate sensors $\mathbf{W}_{\text{all}}$, which is fixed w.r.t. the design $\mathbf{W}$. Consequently, the eigenvalues λ^i_j of the Hessian depend only on $\mathbf{W}$ through the Jacobian $\mathbf{J}$, not on the MAP point. With such a fixed MAP approximation, we can define the approximate EIG as

(3.16)
$$\tilde{\Psi}(\mathbf{W}) := \frac{1}{N_s}\sum_{i=1}^{N_s}\frac{1}{2}\left[\sum_{j=1}^{r}\left(\ln(1+\lambda^i_j(\mathbf{W})) - \frac{\lambda^i_j(\mathbf{W})}{1+\lambda^i_j(\mathbf{W})}\right) + \|\mathbf{m}^i_{\text{map}}(\mathbf{W}_{\text{all}}) - \mathbf{m}_{\text{pr}}\|^2_{\boldsymbol{\Gamma}^{-1}_{\text{pr}}}\right].$$

Since $\|\mathbf{m}^i_{\text{map}}(\mathbf{W}_{\text{all}}) - \mathbf{m}_{\text{pr}}\|^2_{\boldsymbol{\Gamma}^{-1}_{\text{pr}}}$ is independent of $\mathbf{W}$, maximizing $\tilde{\Psi}$ is equivalent to maximizing

$$\tilde{\psi}(\mathbf{W}) := \frac{1}{N_s}\sum_{i=1}^{N_s}\frac{1}{2}\left[\sum_{j=1}^{r}\left(\ln(1+\lambda^i_j(\mathbf{W})) - \frac{\lambda^i_j(\mathbf{W})}{1+\lambda^i_j(\mathbf{W})}\right)\right]. \tag{3.17}$$

3.4. Prior sample point approximation.

With the fixed MAP point approximation, one still needs to solve the optimization problem (3.3) to generate the MAP points $\mathbf{m}^i_{\text{map}}(\mathbf{W}_{\text{all}})$ from the observation data $\boldsymbol{y}_i$, $i = 1, \dots, N_s$. Note that the observation data $\boldsymbol{y}_i$ in (3.2) are provided by first solving the forward model at a sample $\boldsymbol{m}_i$ drawn randomly from the prior, and then extracting the observations from all the candidate sensors. As the number of candidate sensors becomes large, the fixed MAP point $\mathbf{m}^i_{\text{map}}(\mathbf{W}_{\text{all}})$ can recover or approximate the prior sample $\boldsymbol{m}_i$ well if the inverse problem is not severally ill-posed. This provides a rationale to replace the fixed MAP point $\mathbf{m}^i_{\text{map}}(\mathbf{W}_{\text{all}})$ by the prior sample $\boldsymbol{m}_i$. Using this prior sample point approximation, one can completely avoid solving the optimization problem (3.3).

Now we can form our optimization problem as follows: find an optimal design matrix $\mathbf{W}^* \in \mathcal{W}$ such that

$$\mathbf{W}^* = \operatorname*{arg\,max}_{\mathbf{W}\in\mathcal{W}} \tilde{\psi}(\mathbf{W}) = \operatorname*{arg\,max}_{\mathbf{W}\in\mathcal{W}} \frac{1}{N_s}\sum_{i=1}^{N_s}\frac{1}{2}\left[\sum_{j=1}^{r}\left(\ln(1+\lambda^i_j(\mathbf{W})) - \frac{\lambda^i_j(\mathbf{W})}{1+\lambda^i_j(\mathbf{W})}\right)\right], \tag{3.18}$$

where $\lambda^i_1,\ldots,\lambda^i_r$ are the eigenvalues of $\mathbf{W}\hat{\mathbf{H}}_d(\boldsymbol{m}_i)\mathbf{W}^T$.

3.5. Greedy algorithms. With the new optimization object $\tilde{\psi}(\mathbf{W})$, we can develop an online-offline scheme to solve the optimization problem. For the offline stage, we use the randomized SVD in Algorithm 3.1 to compute the low-rank approximation $\hat{\mathbf{H}}_d(\boldsymbol{m}_i)$ of $\mathbf{H}_d(\boldsymbol{m}_i)$ for each data $\mathbf{y}_i$ generated by prior sample $\boldsymbol{m}_i$. For the online stage to choose r sensors out of d candidates that maximize $\tilde{\psi}$ in (3.18), we can treat it as the optimization of set functions under a cardinality constraint set as each design matrix W represents a choice of a subset of candidate sensor location. The set functions usually exhibit properties, such as submodularity (Definition 3.1), that allow for efficient optimization methods [34]. It is well-known that the log-determinant function is a submodular function [36] with the definition of submodularity given as follows.

Definition 3.1. *Let f be a set function on $\mathcal{V}$, i.e., $f:2^{\mathcal{V}}\to\mathbb{R}$. Then f is submodular if for every $A,B\subset\mathcal{V}$ with $A\subset B$ and every $v\in\mathcal{V}\backslash B$, we have $f(A\cup\{v\})-f(A)\geq f(B\cup\{v\})-f(B)$.*

Submodularity is a useful property of set functions with deep theoretical results and applications in combinational optimization and machine learning [13, 45, 62, 50]. The problem of submodular function maximization is a classic NP-hard problem. A general approach to solving it with cardinality constraints is a greedy algorithm. A celebrated result by [50] proves that greedy algorithms can provide a good approximation to the optimal solution of the NP-hard optimization problem despite its simplicity.

The standard greedy algorithm has theoretical guarantees for maximizing submodular functions. Recent work aims to provide theoretical support for its performance for nonsubmodular functions [15]. The greedy algorithm enjoys strong empirical performance and provides near-optimal solutions in practice for many submodular and nonsubmodular functions as shown in [15, 42]. Yet the repeated function evaluations required to compute the objective function in each greedy step may be prohibitive if each function evaluation is expensive.

The following Algorithm 3.2 is the standard greedy algorithm that starts with the empty set S^0, and at each iteration t, the element maximizing the function is added to the chosen sensor set S^t. We denote it as the "standard greedy" algorithm here.

We can see that with each sensor selection, we apply $\mathbf{W}$ on the eigenvector matrix $\mathbf{U}_k$ to actually select the rows of $\mathbf{U}_k$. This can be treated as a row-selection problem to find a subset of rows that captures $\mathbf{U}_k$ as much as possible. This quantity has a natural interpretation in terms of statistical leverage (Definition 3.2) [52], and it has been used extensively to identify

Algorithm 3.2 Standard greedy algorithm.

```
Input: data $\{\mathbf{y}_i\}_{i=1}^{N_s}$ generated from the prior samples $\{\boldsymbol{m}_i\}_{i=1}^{N_s}$, $d$ sensor candidates set $S$, sensor budget $r$, initial set $S^0 = \emptyset$.
for $i = 1, \ldots, N_s$ do
    Compute low-rank approximation $\mathbf{U}_k^i \boldsymbol{\Sigma}_k^i (\mathbf{U}_k^i)^T$ of $\mathbf{H}_d = \hat{\mathbf{J}}_d(\boldsymbol{m}_i)\boldsymbol{\Gamma}_{\text{pr}}\hat{\mathbf{J}}_d^T(\boldsymbol{m}_i)$ by Algorithm 3.1.
end for
for $t = 1, \ldots, r$ do
    $v^* \leftarrow \arg\max_{v \in S \setminus S^{t-1}} \tilde{\psi}(\mathbf{y}, \mathbf{W}, \{\mathbf{U}_k^i, \boldsymbol{\Sigma}_k^i\}_{i=1}^{N_s})$ defined in (3.18), $\mathbf{W}_v$ is the design matrix for the sensor choice $S^{t-1} \cup \{v\}$.
    $S^t \leftarrow S^{t-1} \cup \{v^*\}$.
end for
Output: optimal sensor choice $S^r$.
```

"outlying" or more "informative" data points [22, 42]. Employing this leverage score as a bias toward more informative rows provides a "nice" starting point for the selection in the greedy algorithm [16].

Definition 3.2. *Let $\mathbf{U}_k \in \mathbb{R}^{d\times k}$ contain the eigenvectors corresponding to the k dominant eigenvalues of a $d \times d$ symmetric matrix $\mathbf{A}$ with rank$(\mathbf{A}) \geq k$. Then the (rank-k) leverage score of the ith row of $\mathbf{A}$ is defined as $l_i^k = \|[\mathbf{U}_k]_{i,:}\|^2$ for $i = 1, \ldots, n$, where $[\mathbf{U}_k]_{i,:}$ denotes the ith row of $\mathbf{U}_k$.*

Instead of choosing the sensors one-by-one as in Algorithm 3.2, we propose to use the leverage score information from $\mathbf{U}_k$ as a criterion to select the sensors. We first choose the r rows that have the top-r leverage scores of $\mathbf{U}_k$ as the initial sensor set S^0. At each iteration $t = 1, \ldots, r$, we swap the tth sensor in S^t with one from $S \setminus S^t$ that maximizes the leverage score. Then we set the resulting sensor set S^r as S^0 and repeat the whole process of swapping sensors until it converges such that no sensor is changed. We call this the "swapping greedy algorithm."

3.6. Computation and complexity.

In this section, we present the computation for the MAP point in (3.3) and the low-rank approximation (3.13). Both of these require PDE solves, which overwhelmingly dominate the computational cost. We also present a comparison of computational complexity for different approximations introduced above.

3.6.1. Finding the MAP point.

In the computation of the MAP point, one needs to solve an optimization problem. We use an inexact Newton conjugate gradient (CG) method, which requires computation of the action of the Hessian of the objective (3.3) in a given direction $\hat{m}$, evaluated at a point m, which can be formally written as

$$\mathcal{H}(m)\hat{m} = (D_m^2 \Phi(m, \mathbf{y}) + \mathcal{C}_{\text{pr}}^{-1})\hat{m}. \tag{3.19}$$

The second term can be evaluated by (2.16) as

$$\mathcal{C}_{\text{pr}}^{-1}\hat{m} \approx \boldsymbol{\Gamma}_{\text{pr}}^{-1}\hat{\boldsymbol{m}} = \mathbf{A}\mathbf{M}^{-1}\mathbf{A}\hat{\boldsymbol{m}}, \tag{3.20}$$

Algorithm 3.3 Swapping greedy algorithm.

```
Input: data $\{\mathbf{y}_i\}_{i=1}^{N_s}$ generated from the prior samples $\{\boldsymbol{m}_i\}_{i=1}^{N_s}$, $d$ sensor candidates
  set $S$, sensor budget $r$.
for $i = 1, \dots, N_s$ do
  Compute low-rank approximation $\mathbf{U}_k^i \mathbf{\Sigma}_k^i (\mathbf{U}_k^i)^T$ of $\mathbf{H}_d = \hat{\mathbf{J}}_d(\boldsymbol{m}_i)\mathbf{\Gamma}_{\mathrm{pr}}\hat{\mathbf{J}}_d^T(\boldsymbol{m}_i)$ by
    Algorithm 3.1.
end for
Compute the initial guess sensor choice $S^* = \{s_1, \dots, s_r\} \subset S$ based on leverage scores
  of $\sum_{i=1}^{N_s} \mathbf{U}_k^i$. $S^0 = \{\emptyset\}$.
while $S^* \neq S^0$ do
  $S^0 \leftarrow S^*$
  for $t = 1, \dots, r$ do
    $v^* \leftarrow \arg\max_{v \in \{s_t\} \cup (S \setminus S^{t-1})} \tilde{\psi}(\mathbf{y}, \mathbf{W}, \{\mathbf{U}_k^i, \mathbf{\Sigma}_k^i\}_{i=1}^{N_s})$ defined in (3.18),
    $\mathbf{W}_v$ is the design matrix for the sensor choice $S^{t-1} \setminus \{s_t\} \cup \{v\}$.
    $S^t \leftarrow (S^{t-1} \setminus \{s_t\}) \cup \{v^*\}$.
  end for
  $S^* \leftarrow S^r$.
end while
Output: optimal sensor choice $S^*$.
```

which requires the solution of a linear system with mass matrix $\mathbf{M}$. The first term of (3.19) can be evaluated by the Lagrange method. The potential $\Phi(m, \mathbf{y})$ can be explicitly written as $\Phi(u(m)) = \frac{1}{2}||\mathcal{B}(u) - \mathbf{y}||^2_{\mathbf{\Gamma}_{\mathrm{n}}^{-1}}$. We introduce the Lagrangian

$$\mathcal{L}(u, m, v) = \Phi(u) + r(u, m, v), \tag{3.21}$$

where v is the Lagrangie multiplier, and $r(u, m, v)$ is the weak form of the forward PDE (2.2). By setting variations of $\mathcal{L}$ w.r.t. v as zero, we obtain the forward problem (2.2), which can be equivalently written as follows: find $u \in \mathcal{V}$ such that

$$\langle \tilde{v}, \partial_v r(u, m, v)\rangle = 0 \quad \forall \tilde{v} \in \mathcal{V}. \tag{3.22}$$

By setting variations of $\mathcal{L}$ w.r.t. u to zero, we obtain the adjoint problem: find $v \in \mathcal{V}$ such that

$$\langle \tilde{u}, \partial_u r(u, m, v)\rangle = -\langle \tilde{u}, \partial_u \Phi(u)\rangle \quad \forall \tilde{u} \in \mathcal{V}, \tag{3.23}$$

where $\langle \tilde{u}, \partial_u \Phi(u)\rangle = (\mathcal{B}(u) - \mathbf{y})^T \mathbf{\Gamma}_{\mathrm{n}}^{-1} \mathcal{B}(\tilde{u})$. The gradient of Φ w.r.t. m can be evaluated as

$$\langle \tilde{m}, D_m \Phi(m, \mathbf{y})\rangle = \langle \tilde{m}, \partial_m \mathcal{L}(u, m, v)\rangle = \langle \tilde{m}, \partial_m r(u, m, v)\rangle. \tag{3.24}$$

To compute the Hessian action $D_m^2 \Phi\, \hat{m}$, we introduce another Lagrangian $\mathcal{L}^H$ that combines the gradient (3.24) and adjoint (3.23) problems enforced via the Lagrange multipliers $\hat{u}$ and $\hat{v}$, the constraints (3.22) and (3.23) as

$$\mathcal{L}^H(u, m, v, \hat{u}, \hat{m}, \hat{v}) = \langle \hat{m}, \partial_m r(u, m, v)\rangle + \langle \hat{v}, \partial_v r(u, m, v)\rangle + \langle \hat{u}, \partial_u r(u, m, v) + \partial_u \Phi(u)\rangle. \tag{3.25}$$

By setting variations of $\mathcal{L}^H$ w.r.t. v to zero, we obtain the incremental forward problem: find $\hat{u} \in \mathcal{V}$ such that

$$\langle \tilde{v}, \partial_{uv} r\, \hat{u}\rangle = -\langle \tilde{v}, \partial_{mv} r\, \hat{m}\rangle \quad \forall \tilde{v} \in \mathcal{V}, \tag{3.26}$$

where $\partial_{uv} r : \mathcal{V} \to \mathcal{V}'$ and $\partial_{mv} r : \mathcal{M} \to \mathcal{V}'$ are linear operators. By setting variations of $\mathcal{L}^H$ w.r.t. u to zero, we obtain the incremental adjoint problem: find $\hat{v} \in \mathcal{V}$ such that

$$\langle \tilde{u}, \partial_{vu} r\, \hat{v}\rangle = -\langle \tilde{u}, \partial_{mu} r\, \hat{m}\rangle - \langle \tilde{u}, \partial_{uu} r\, \hat{u}\rangle - \langle \tilde{u}, \partial_{uu} \Phi\, \hat{u}\rangle \quad \forall \tilde{u} \in \mathcal{V}, \tag{3.27}$$

where $\partial_{vu} r : \mathcal{V} \to \mathcal{V}'$, $\partial_{mu} r : \mathcal{M} \to \mathcal{V}'$, and $\partial_{uu} r : \mathcal{V} \to \mathcal{V}'$ are linear operators, and $\langle \tilde{u}, \partial_{uu} \Phi\, \hat{u}\rangle = \mathcal{B}(\hat{u})^T \boldsymbol{\Gamma}_{\rm n}^{-1} \mathcal{B}(\tilde{u})$. To this end, the Hessian action $D^2_m \Phi\, \hat{m}$ can be evaluated as

$$\langle \tilde{m}, D^2_m \Phi\, \hat{m}\rangle = \langle \tilde{m}, \partial_m \mathcal{L}^H\, \hat{m}\rangle = \langle \tilde{m}, \partial_{mm} r\, \hat{m} + \partial_{vm} r\, \hat{v} + \partial_{um} r\, \hat{u}\rangle, \tag{3.28}$$

where $\partial_{mm} r : \mathcal{M} \to \mathcal{M}'$, $\partial_{vm} r : \mathcal{V} \to \mathcal{M}'$, and $\partial_{um} r : \mathcal{V} \to \mathcal{M}'$ are linear operators. Therefore, at m, after solving one forward problem (3.22) and one adjoint problem (3.23), the Hessian action $D^2_m \Phi\, \hat{m}$ at each $\hat{m}$ requires the solution of one incremental forward problem (3.26) and one incremental adjoint problem (3.27), i.e., two linearized PDE solves. We solve all the above PDEs by a finite element method in the subspace $\mathcal{V}_{n_u} \subset \mathcal{V}$, and $\mathcal{M}_n \subset \mathcal{M}$.

3.6.2. Computing the H_d action requred for its low-rank approximation.

In the computation of the low-rank approximation (3.13) by Algorithm 3.1, we need to perform the actions $\mathbf{H}_d \boldsymbol{\Omega}$ and $\mathbf{H}_d \boldsymbol{Q}$, where $\mathbf{H}_d$ is defined in (3.13). We next present the computation of the action $\mathbf{H}_d \hat{\boldsymbol{z}}$ in an arbitrary direction $\hat{\boldsymbol{z}} \in \mathbb{R}^d$. By definition, we have

$$\mathbf{H}_d \hat{\boldsymbol{z}} = \hat{\mathbf{J}}_d \boldsymbol{\Gamma}_{\rm pr} \hat{\mathbf{J}}_d^T \hat{\boldsymbol{z}} = (\boldsymbol{\Gamma}_{\rm n}^d)^{-\frac{1}{2}} \mathbf{J}_d \boldsymbol{\Gamma}_{\rm pr} \mathbf{J}_d^T (\boldsymbol{\Gamma}_{\rm n}^d)^{-\frac{1}{2}} \hat{\boldsymbol{z}} = (\boldsymbol{\Gamma}_{\rm n}^d)^{-\frac{1}{2}} \mathbf{J}_d \mathbf{A}^{-1} \mathbf{M} \mathbf{A}^{-1} \mathbf{J}_d^T (\boldsymbol{\Gamma}_{\rm n}^d)^{-\frac{1}{2}} \hat{\boldsymbol{z}}, \tag{3.29}$$

which involves the computation of the actions of the Jacobian $\mathbf{J}_d$ and its transpose $\mathbf{J}_d^T$, and two solves of a linear system with stiffness matrix $\mathbf{A}$. We first consider the action of $\mathbf{J}_d^T \boldsymbol{z}$ in direction $\boldsymbol{z} = (\boldsymbol{\Gamma}_{\rm n}^d)^{-\frac{1}{2}} \hat{\boldsymbol{z}}$. By definition of the Jacobian in (3.6) and $\mathcal{F}(m) = \mathcal{B}_d(u(m))$, we have $\mathbf{J}_d^T \boldsymbol{z} = (D_{\boldsymbol{m}} \boldsymbol{u}(\boldsymbol{m}))^T \mathbf{B}_d^T \boldsymbol{z}$, where the jth column $(\mathbf{B}_d)_j = \mathcal{B}_d(\psi_j) \in \mathbb{R}^d$ for the basis functions $\{\psi_j\}_{j=1}^{n_u}$ in approximating the state $u = \sum_{j=1}^{n_u} u_j \psi_j$, and $\boldsymbol{u} = (u_1, \dots, u_{n_u})^T \in \mathbb{R}^{n_u}$ is the coefficient vector. By taking the variation of the forward problem in the form (3.22) w.r.t. m, and noting that u depends on m, we have

$$0 = \langle \tilde{v}, \partial_{vu} r\, D_m u(m) + \partial_{vm} r\rangle \quad \forall v \in \mathcal{V}. \tag{3.30}$$

Let $\mathbf{R}_{vu}$ and $\mathbf{R}_{vm}$ denote the matrices corresponding to the finite element discretization of $\partial_{vu} r$ and $\partial_{vm} r$ above, respectively. We formally obtain

$$D_{\boldsymbol{m}} \boldsymbol{u}(\boldsymbol{m}) = -\mathbf{R}_{vu}^{-1} \mathbf{R}_{vm}, \tag{3.31}$$

so that $\mathbf{J}_d^T \boldsymbol{z}$ can be evaluated by solving the linear system $\mathbf{R}_{vu}^T \boldsymbol{w} = \mathbf{B}_d^T \boldsymbol{z}$ for $\boldsymbol{w} \in \mathbb{R}^{n_u}$, which has the same coefficient matrix $\mathbf{R}_{vu}^T = \mathbf{R}_{uv}$ as in the discrete incremental forward problem (3.26), and performing the matrix-vector product $-\mathbf{R}_{vm}^T \boldsymbol{w}$. Similarly the Jacobian action $\mathbf{J}_d \boldsymbol{n} = \mathbf{B}_d D_{\boldsymbol{m}} \boldsymbol{u}(\boldsymbol{m}) \boldsymbol{n}$ for $\boldsymbol{n} = \mathbf{A}^{-1} \mathbf{M} \mathbf{A}^{-1} \mathbf{J}_d^T \boldsymbol{z}$ can be evaluated by first performing the matrix-vector product $\mathbf{R}_{vm} \boldsymbol{n}$, then solving the linear system $\mathbf{R}_{vu} \boldsymbol{w} = \mathbf{R}_{vm} \boldsymbol{n}$ for $\boldsymbol{w} \in \mathbb{R}^{n_u}$, which has the same matrix $\mathbf{R}_{vu}$ as in the discrete incremental adjoint problem (3.27), and finally performing the matrix-vector product $-\mathbf{B}_d \boldsymbol{w}$. In summary, after solving one forward problem (3.22) for u and one adjoint problem (3.23) for v, each action $\mathbf{H}_d \hat{\boldsymbol{z}}$ consists of solving four linearized PDEs, one with matrix $\mathbf{R}_{vu}$, one with $\mathbf{R}_{uv}$, and two with $\mathbf{A}$.

Table 1

Computational complexity in terms of the number of $\mathbf{H}$ *actions* (3.28) *for finding MAP point and* $\mathbf{H}_d$ *actions* (3.29) *for low-rank approximation.* N_s*: # data,* N_{kl}*: # EIG evaluations,* N_{nt}*: # Newton iterations,* N_{cg}*: # CG iterations,* k*: # rank of the low-rank approximation* (3.13)*,* p*: an oversampling factor in Algorithm* 3.1*.*

# H or H_d actions	Exact MAP point	Fixed MAP point	Prior sample point
Finding MAP point (by Newton CG)	$N_{kl} \times N_s \times N_{nt} \times N_{cg}$	$N_s \times N_{nt} \times N_{cg}$	0
Low-rank approximation (by RSVD Algorithm 3.1)	$N_{kl} \times N_s \times 2(k+p)$	$N_s \times 2(k+p)$	$N_s \times 2(k+p)$

3.6.3. Computational complexity.

The cost of solving the OED problem is overwhelmingly dominated by the costs of PDE solves needed to form the actions of either $\mathbf{H}$ or $\mathbf{H}_d$ on given vectors. Recall that a pair of linearized forward/adjoint PDE solves, in addition to two elliptic PDE solves representing the prior, are required to form actions with $\mathbf{H}$ or $\mathbf{H}_d$. The remaining costs, which involve linear algebra, are negligible relative to those PDE solves, for anything other than small model problems. Thus, in the section, we characterize the complexity of solving the OED problem under the three stages of approximation (exact MAP point, fixed MAP point, prior sample point) using the number of $\mathbf{H}$ or $\mathbf{H}_d$ actions as a measure of cost.

Suppose we need to evaluate N_{kl} times of the objective function of the OED problem, i.e., the KL divergence (3.1) for each of the N_s training data $\{\mathbf{y}_i\}_{i=1}^{N_s}$. Each time corresponds to a different choice of sensor locations to find the optimal design by the greedy algorithms. Assume that to find the MAP point for each training data, we need N_{nt} Newton iterations and an average of N_{cg} CG iterations for each Newton iteration. Then, in total, we need $N_{kl} \times N_s \times N_{nt} \times N_{cg}$ Hessian actions (3.19) to compute the exact MAP points, $N_s \times N_{nt} \times N_{cg}$ $\mathbf{H}_d$ actions to compute the fixed MAP points, and 0 Hessian actions if the prior points are used. Assume that on average the rank k with an oversampling factor $p = 10$ is used in the low-rank approximation in Algorithm 3.1. Thus, for the low-rank approximation of $\mathbf{H}_d$, in total we need $N_{kl} \times N_s \times 2(k+p)$ $\mathbf{H}_d$ actions (3.29) for the case of an exact MAP point, $N_s \times 2(k+p)$ $\mathbf{H}_d$ actions for the case of a fixed MAP point, and $N_s \times 2(k+p)$ $\mathbf{H}_d$ actions for case of a prior point. We summarize the computational complexity in terms of $\mathbf{H}_d$ actions in Table 1.

3.7. Special case of linear Bayesian inverse problems.

For the linear Bayesian inverse problems, by which we mean that the parameter-to-observable map $\mathbf{F}(\boldsymbol{m})$ is linear w.r.t. the parameter $\boldsymbol{m}$, where for a specific design W, we have $\mathbf{F} = \mathbf{W}\mathbf{F}_d$. The KL divergence [1] has a closed form expression defined in (2.8). Employing the discretization of subsection 2.4, the EIG reduces to [1]

$$\Psi = \frac{1}{2}\text{logdet}\left(\mathbf{I} + \widetilde{\mathbf{H}}_m\right), \tag{3.32}$$

where $\widetilde{\mathbf{H}}_m = \mathbf{\Gamma}_{\text{pr}}^{\frac{1}{2}}\mathbf{F}^T\mathbf{\Gamma}_{\text{n}}^{-1}\mathbf{F}\mathbf{\Gamma}_{\text{pr}}^{\frac{1}{2}}$. With no need of further approximation of MAP point, we can employ our method to present an efficient approximation of the discrete EIG given in (3.32) and establish an error estimate for the approximation in Theorem 3.3 with the proof in Appendix A.

Theorem 3.3. *Let* $\mathbf{H}_d := \hat{\mathbf{F}}_d \mathbf{\Gamma}_{\text{pr}} \hat{\mathbf{F}}_d^T \in \mathbb{R}^{d\times d}$ *with* $\hat{\mathbf{F}}_d = (\mathbf{\Gamma}_{\text{n}}^d)^{-\frac{1}{2}} \mathbf{F}_d$*, where* $\mathbf{\Gamma}_{\text{n}}^d$ *is defined in* (2.11)*. We compute a low-rank decomposition of* $\mathbf{H}_d$ *as*

$$\hat{\mathbf{H}}_d = \mathbf{U}_k \mathbf{\Sigma}_k \mathbf{U}_k^T, \tag{3.33}$$

where $(\mathbf{\Sigma}_k, \mathbf{U}_k)$ *represent the* k *dominant eigenpairs, with* $\mathbf{\Sigma}_k = diag(\lambda_1, \dots, \lambda_k)$ *for eigenvalues* $\lambda_1 \geq \cdots \geq \lambda_k$*. Moreover, let* $\hat{\Psi}$ *denote an approximate EIG defined as*

$$\hat{\Psi}(\mathbf{W}) := \frac{1}{2}\text{logdet}\left(\mathbf{I}_{r\times r} + \mathbf{W}\hat{\mathbf{H}}_d\mathbf{W}^T\right). \tag{3.34}$$

Then we have

$$0 \leq \Psi(\mathbf{W}) - \hat{\Psi}(\mathbf{W}) \leq \frac{1}{2}\sum_{i=k+1}^{d} \log(1+\lambda_i), \tag{3.35}$$

where $\lambda_{k+1}, \dots, \lambda_d$ *are the trailing eigenvalues of* $\mathbf{H}_d$*.*

We can then apply our online-offline scheme to efficiently evaluate $\hat{\Psi}$ and solve the optimization problem with the greedy algorithms as in subsection 3.5.

4. Numerical results.

In this section, we present numerical results of both a linear advection-diffusion problem for the inversion of the initial condition and a nonlinear Poisson problem for the inversion of a diffusion coefficient. We demonstrate that our proposed approximations and the greedy algorithm are effective and efficient, and our method is scalable w.r.t. the number of training data points, the number of candidate sensors, and the parameter dimension.

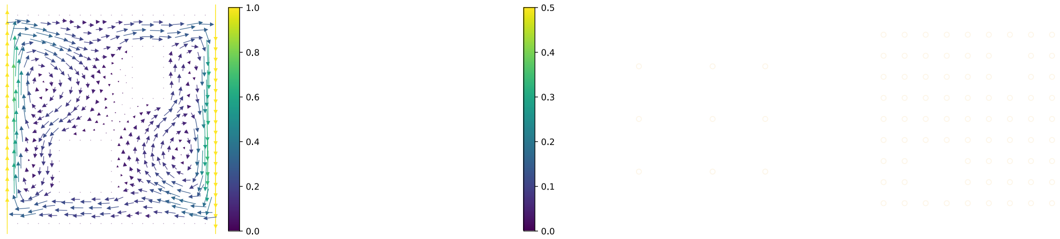
4.1. A linear Bayesian inverse problem.

In this example, we consider inversion of the initial condition of an advection-diffusion problem given pointwise observations of the state at certain sensor locations and certain times. The forward problem is given by

$$u_t - k\Delta u + \boldsymbol{v}\cdot\nabla u = 0 \text{ in } \mathcal{D}\times(0,T), \tag{4.1a}$$
$$u(\cdot,0) = m \text{ in } \mathcal{D}, \tag{4.1b}$$
$$k\nabla u\cdot n = 0 \text{ on } \partial\mathcal{D}\times(0,T), \tag{4.1c}$$

where $\mathcal{D} = (0,1)^2 \subset \mathbb{R}^2$ is the open and bounded domain with boundary $\partial\mathcal{D}$ depicted in Figure 1, $k > 0$ is the diffusion coefficient, and $T > 0$ is the final time. In our numerical experiments, we choose $k = 0.001$. The velocity field $\boldsymbol{v} \in \mathbb{R}^2$ is obtained by solving the following steady-state Navier–Stokes equation with the side walls driving the flow:

$$-\frac{1}{\text{Re}}\Delta\boldsymbol{v} + \nabla q + \boldsymbol{v}\cdot\nabla\boldsymbol{v} = 0 \text{ in } \mathcal{D}, \tag{4.2a}$$
$$\nabla\cdot\boldsymbol{v} = 0 \text{ in } \mathcal{D}, \tag{4.2b}$$
$$\boldsymbol{v} = \boldsymbol{g} \text{ on } \partial\mathcal{D}. \tag{4.2c}$$



(a) velocity field $\boldsymbol{v}$

(b) The "true" initial condition

(c) 9 candidates

(d) 75 candidates

Figure 1. *The computational domain $\mathcal{D}$ for a two-dimensional problem is $[0,1]^2$ with two rectangular blocks ($[0.25,0.5]\times[0.15,0.4]$, $[0.6,0.75]\times[0.6,0.85]$) removed.* (a) *The velocity field obtained by solving* (4.2). (b) *The "true" initial condition m_{true} of parameter m.* (c) *The state field at time $T=4$ obtained by solving* (4.1) *with* 9 *candidate sensor locations.* (d) *The state field at time $T=4$ obtained by solving* (4.1) *with* 75 *candidate sensor locations.*

Here q represents the pressure field, and the Reynolds number Re is set at 50. The Dirichlet boundary data $\boldsymbol{g}\in\mathbb{R}^2$ is prescribed as $\boldsymbol{g}=(0,1)$ on the left wall of the domain, $\boldsymbol{g}=(0,-1)$ on the right wall, and $\boldsymbol{g}=(0,0)$ elsewhere.

We use a Galerkin finite element method with piecewise linear elements for spatial discretization of the forward and adjoint problems, which results in $n=2023$ spatial degrees of freedom for the parameter m and state variable u. We use the implicit Euler method for temporal discretization with $N_t=40$ time steps for the final time $T=4$. We consider a Gaussian prior $\mu_{\text{pr}}=\mathcal{N}(m_{\text{pr}},\mathcal{C}_{\text{pr}})$ for the parameter m. The covariance $\mathcal{C}_{\text{pr}}=\mathcal{A}^{-2}$ is given by the square of the inverse of differential operator $\mathcal{A}=-\gamma\Delta+\delta I$ with Laplacian Δ and identity I, equipped with Robin boundary condition $\gamma\nabla m\cdot\mathbf{n}+\beta m$ on $\partial\mathcal{D}$, where $\gamma,\delta>0$ control the correlation length and variance of the prior distribution. The Robin coefficient β is chosen as in [32] to reduce boundary artifacts. For our numerical test, we choose $m_{\text{pr}}=0.25$, $\gamma=1,\delta=8$, and set a "true" initial condition $m_{\text{true}}=\min(0.5,\exp(-100\|x-[0.35,0.7]\|^2)$. The velocity field and initial condition are shown in Figure 1 (left two), to generate the observation data at the final time, as shown in Figure 1 (right two). We compute the low-rank approximation (3.33) and its eigenvalues are displayed in Figure 2 with the increasing number of parameter dimensions and data dimensions (number of candidate sensor locations). We can see eigenvalues decay rapidly, over five orders of magnitude in the first 20 eigenvalues, independent of data dimension and parameter dimension. This shows that we only need a limited number of PDE solves for low-rank approximations of $\hat{\mathbf{H}}_d$ to evaluate EIG for different designs, scalable w.r.t. the number of PDE solves.

In the first test, we use a small number of candidate sensors and compare the design obtained by the greedy algorithms with the optimal design by brute-force search to show the efficacy of the greedy algorithms. Specifically, we use a grid of $d=9$ candidate sensor locations $\{x_i\}_{i=0}^{9}$ ($x_i\in\{0.2,0.55,0.8\}\times\{0.25,0.5,0.75\}$) as shown in Figure 1(c) with the goal of choosing $r=2,3,4,5,6,7,8$ sensors at the finial time. We run the two greedy algorithms, Algorithms 3.2 and 3.3, as well as a brute-force search of all possible designs ($\frac{9!}{r!(9-r)!}$) to find the optimal design. In the evaluation of the approximate EIG (3.34), we do not need the low-rank approximation of $\mathbf{H}_d$ here as in Theorem 3.3 since it is a small (9×9) matrix that

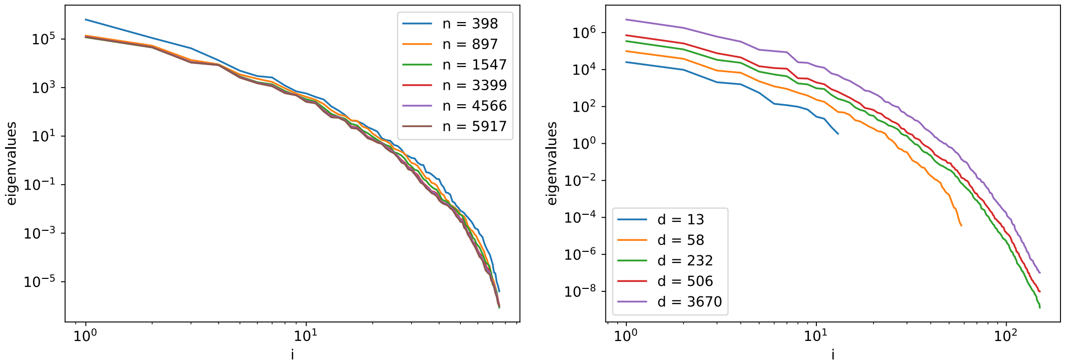
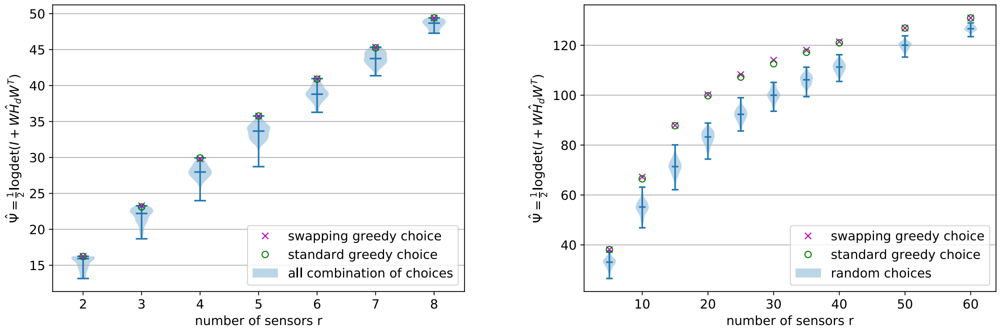


Figure 2. *Decay of the eigenvalues of Hessian $\hat{H}_d$ in (3.33) with the increasing number of parameter dimensions n (left) and candidate sensor locations d (right).*



(a) 9 candidate sensors

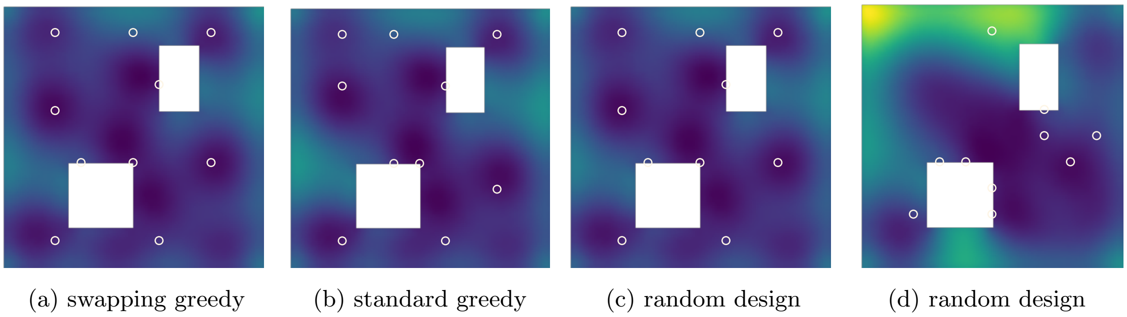
(b) 75 candidate sensors

Figure 3. *EIG with the increasing number of sensors. Blue filled areas represent the probability distributions for all the designs with lines at the minimum, maximum, and median.*

we can easily compute by solving 9 incremental forward and 9 incremental adjoint problems in directions e_j of dimension 9 with the jth element as one and others as zeros, $j = 1, \ldots, 9$. The values of the approximate EIG by the two algorithms and at all possible designs are shown in Figure 3(a).

We can see that for $r = 2, 3, 5, 6, 7, 8$ the swapping greedy algorithm finds the optimal design for all r but $r = 4$ with the second best, while the standard greedy algorithm finds the optimal for $r = 4, 5, 8$, the second-best for $r = 6, 7$, the third for $r = 2$, and the fifth for $r = 3$. Although the optimal designs are not found in all cases, those chosen by the two greedy algorithms are quite close to the optimal.

In the second test, we consider a grid of $d = 75$ candidate sensor locations as shown in Figure 1(d) with the goal of choosing r sensor locations from 5 to 65 in increments of 5. We randomly draw 200 different designs from the candidate sensors and compute their approximate EIG and compare them with the ones chosen by two greedy algorithms as shown in Figure 3(b), from which we can see that both greedy algorithms find the designs better than all the random choices, and the new swapping greedy algorithm we propose always gives higher (better) or at least equal values.



(a) swapping greedy (b) standard greedy (c) random design (d) random design

Figure 4. *Pointwise variance of the posterior at designs chosen by swapping greedy algorithm, the standard greedy algorithm, and two random designs with* 10 *sensors. The brighter region corresponds to larger variance. Compared with the optimal design chosen by the swapping greedy algorithm, the standard greedy algorithm and the two random designs lead to* $7\%, 30\%, 53\%$ *increase in the averaged variance, respectively.*

Moreover, the advantage of our swapping greedy algorithm can also be illustrated by reduced pointwise posterior variance indicated in Figure 4 compared to the standard greedy algorithm and two random designs with the same number of sensors.

4.2. A nonlinear Bayesian inverse problem. In this problem, we consider a log-normal diffusion forward model as follows:

$$-\nabla \cdot (\exp(m)\nabla u) = f \text{ in } \mathcal{D}, \tag{4.3a}$$

$$u = g \text{ on } \Gamma_D, \tag{4.3b}$$

$$\exp(m)\nabla u \cdot \mathbf{n} = h \text{ on } \Gamma_N, \tag{4.3c}$$

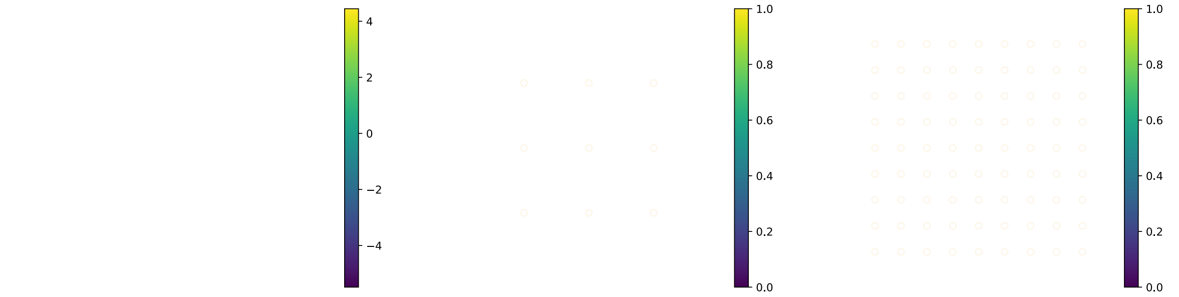
where $\mathcal{D} \subset \mathbb{R}^2$ is an open, bounded domain with sufficiently smooth boundary $\Gamma = \Gamma_D \cup \Gamma_N$ with Dirichlet and Neumann boundaries $\Gamma_D \cap \Gamma_N = \emptyset$ and data $g \in \mathcal{H}^{1/2}(\Gamma_d)$ and $h \in L^2(\Gamma_N)$, respectively. The state variable $u \in \mathcal{V}_g = \{v \in \mathcal{H}^1(\mathcal{D}) : v|_{\Gamma_D} = g\}$. $f \in L^2(\mathcal{D})$ is a source term. We consider a Gaussian prior for the parameter $m \in \mathcal{H}^1(\mathcal{D})$, i.e., $m \sim \mu_{\text{pr}} = \mathcal{N}(m_{\text{pr}}, \mathcal{C}_{\text{pr}})$ with mean m_{pr} and covariance $\mathcal{C}_{\text{pr}} = \mathcal{A}^{-2}$, where $\mathcal{A}$ is a differential operator given by

$$\mathcal{A}m = \begin{cases} -\gamma\nabla \cdot (\Theta\nabla m) + \delta m & \text{in } \mathcal{D}, \\ \Theta\nabla m \cdot \mathbf{n} + \beta m & \text{on } \partial\mathcal{D}, \end{cases} \tag{4.4}$$

where $\beta \approx \sqrt{\gamma\delta}$ is the optimal Robin coefficient derived in [32] to minimize boundary artifacts and Θ is an symmetric positive definite and anisotropic matrix of the form

$$\Theta = \begin{bmatrix} \theta_1 \sin(\alpha)^2 & (\theta_1 - \theta_2)\sin(\alpha)\cos(\alpha) \\ (\theta_1 - \theta_2)\sin(\alpha)\cos(\alpha) & \theta_2 \cos(\alpha)^2 \end{bmatrix}. \tag{4.5}$$

In our numerical experiment, we set the prior mean to be zero, $\gamma = 0.04, \delta = 0.2, \theta_1 = 2, \theta_2 = 0.5, \alpha = \pi/4$. For the forward problem, we consider the domain $\mathcal{D} = (0,1) \times (0,1)$, no source term (i.e., $f = 0$), and no normal flux on $\Gamma_N = \{0,1\} \times (0,1)$, i.e., imposing the homogeneous Neumann condition $\exp(m)\nabla u \cdot \mathbf{n} = 0$. The Dirichlet boundary $\Gamma_D = (0,1) \times \{0,1\}$ with boundary condition $u = 1$ on $(0,1) \times \{1\}$ and $u = 0$ on $(0,1) \times \{0\}$. We draw a sample from the



(a) The synthetic "true" parameter (b) 9 candidate sensor locations (c) 81 candidate sensor locations

Figure 5. *The computational domain $\mathcal{D} = (0,1) \times (0,1)$ with no source term and no normal flux on $\Gamma_N = \{0,1\} \times (0,1)$.* (a) *The "true" parameter m_{true} of parameter m.* (b) *The state field at the "true" parameter with 9 candidate sensor locations.* (c) *The state field at the "true" parameter with 81 candidate sensor locations.*

prior and use it as the "true" parameter field m_{true} as shown Figure 5(a). We use quadratic finite elements for the discretization of the state and adjoint variables and use linear elements for the parameter. The degrees of freedom for the state and parameter are $n_u = 4225$ and $n = 1089$, respectively.

4.2.1. Effectiveness of the approximations. To reduce the computational cost for the nonlinear inverse problems, we introduced the Laplace approximation with low-rank decomposition, fixed MAP point approximation, and prior sample point approximation in section 3. To investigate their effectiveness, we consider their (sample) correlation at the same design. A high correlation (with correlation coefficient close to 1) of the approximate EIG values by two different approximations implies that the optimal design obtained by one approximation is likely to be close to optimal for the other approximation.

In the test, we use the grid of $d = 81$ candidate sensor locations as shown in Figure 5(c) with the goal of choosing $r = 10$ sensor locations. We generate 200 random designs and compute the EIG of each design by a DLMC method as the reference. Then we compute their approximate EIG by the Laplace approximation (LA-EIG) and its further approximation with fixed map point shown in Figure 6 (left). We can see that the correlation between DLMC-EIG and LA-EIG is 0.9752, and the maximum of DLMC-EIG is also the maximum of LA-EIG. The correlation between DLMC-EIG and LA-EIG with fixed map point is 0.9453, and the maximum of LA-EIG with fixed map point is the second maximum of DLMC-EIG. Although it is not the maximum, it gives almost the same EIG value as the maximum. We can observe closely the relation between LA-EIG and LA-EIG with fixed MAP point in Figure 6 (middle) with a correlation 0.9525 and that their optimal choices have almost the same LA-EIG values. Figure 6 (right) illustrates the correlation between $\tilde{\psi}$ computed with the fixed MAP point and with the prior sample at which to evaluate Hessian. We can see close to 1 correlation and that the optimal choices with the fixed map point and the prior sample point are the same.

4.2.2. Numerical results. We first use a grid of $d = 9$ candidate sensor locations with the goal of choosing $r = 2,3,4,5,6,7,8$. As we can see in Figure 7(a), the standard greedy and swapping greedy algorithms give the same optimal design for all the cases and they are

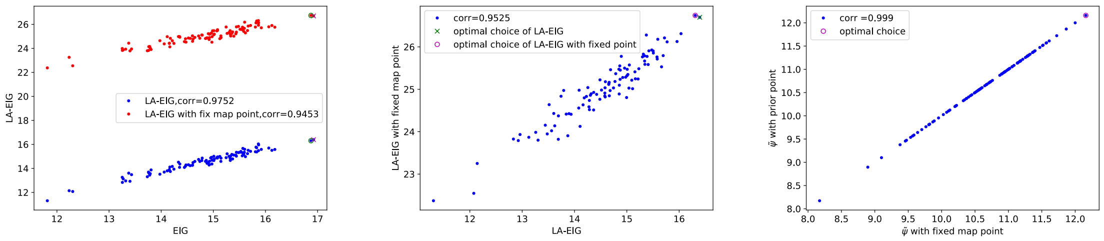
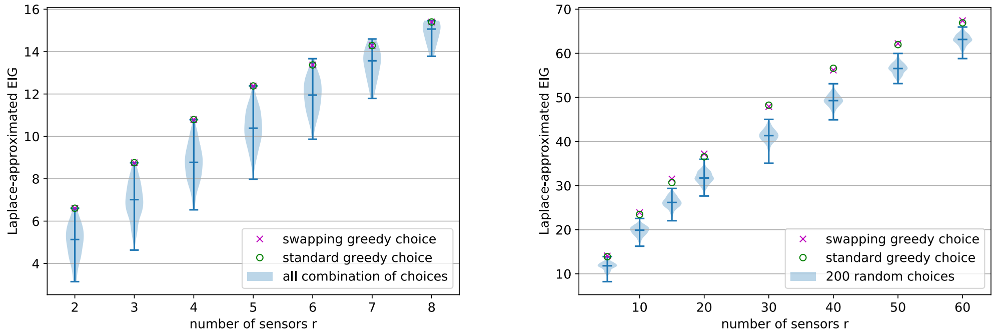


Figure 6. *Correlation between EIG and its approximations. Each blue or red point represents one random design of* 10 *sensors among* 81 *candidates. Left: DLMC-EIG vs. LA-EIG and LA-EIG with fixed MAP point approximation. Middle: LA-EIG and LA-EIG with fixed MAP point approximation. Right:* $\tilde{\Psi}$ *with fixed MAP point approximation vs. LA-EIG with prior sample point approximation.*



(a) 9 candidate sensor locations

(b) 81 candidate sensor locations

Figure 7. *Laplace approximate EIG with the increasing number of sensors. Blue filled areas represent the probability distributions for all the designs with lines at the minimum, maximum, and median.*

the actual optimal one among all possible choices for $r = 2, 3, 5, 8$, second-best for $r = 4, 7$, and third for $r = 6$. We remark that by evaluating the Hessian at the prior sample point, the computational cost is significantly reduced as analyzed in subsection 3.6. Despite the fact that the optimal choice by the approximation might not be optimal for LA-EIG, we can still find the ones close to the best in all the cases.

Then we consider 81 candidate sensor locations shown in Figure 5(c). We randomly draw 200 different designs from the candidate sensors and compute their LA-EIG, and compare them with LA-EIG of the choices by two greedy algorithms. Figure 7(b) illustrates that the designs chosen by both greedy algorithms are much better than all the random choices, and the swapping greedy algorithm is mostly better or equal to the standard greedy algorithm.

To illustrate the stability of our method, we compare the greedy choices with 200 random designs with the increasing number of training data N_s and increasing number of candidate sensor locations d (data dimension) in Figure 8. We see that the swapping greedy algorithm always finds better designs than the standard greedy algorithm, and much better than the random choices.

4.2.3. Scalability. As analyzed in subsection 3.6, the computational complexity in terms of the number of PDE solves critically depends on the rank k in the low-rank approximation

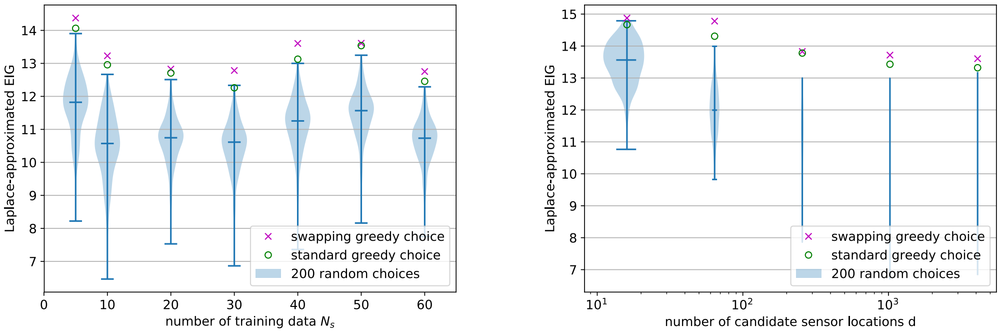


Figure 8. *LA-EIG for choosing* 5 *sensors at optimal (by the swapping and standard greedy algorithms) and random designs with increasing number of training data (left) and increasing number of candidates (right). Blue filled areas represent the probability distributions of LA-EIG for* 200 *random designs.*

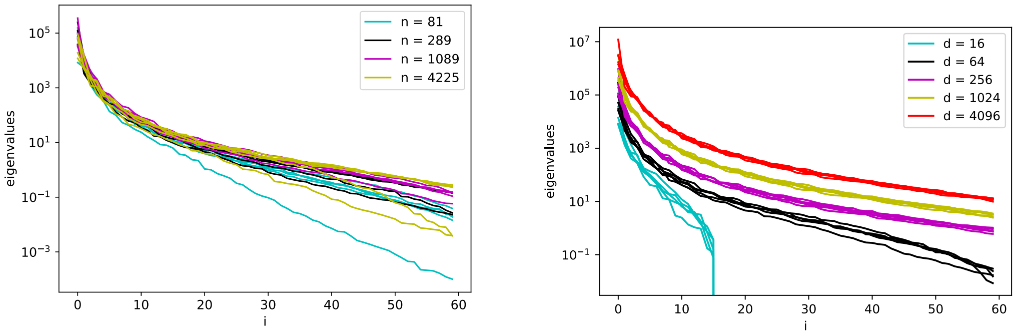


Figure 9. *The decay of eigenvalues of the $\hat{H}_d$ with increasing number of parameter dimension n and candidate sensor locations d. Multiple lines with the same color represent different training data cases.*

of the Hessian $\hat{\mathbf{H}}_d$. In this section, we investigate the dependence of k on the parameter dimension and data dimension (the number of candidate sensor locations). The decay of the eigenvalues of the Hessian is shown in Figure 9 with increasing parameter dimension (left) and data dimension (right). From the similarity of the decay rates of the eigenvalues in the left part of the figure, we can conclude that our algorithm is scalable w.r.t. the parameter dimension in the sense that k is essentially independent of the parameter dimension, once the parameter filed is sufficiently resolved. The similarity of the decay rates of the eigenvalues of the Hessian in the right part of the figure suggests that k is only weakly dependent on the data dimension.

5. Conclusion.

We have developed a fast and scalable computational framework for both linear and nonlinear Bayesian OED problems governed by PDEs. It exploits the low-rank structure of the prior-preconditioned data misfit Hessian, the dominant data subspace information from the Jacobian of the parameter-to-observable map, the Laplace approximation of the posterior in nonlinear Bayesian inverse problems, and the approximation of the MAP point by the prior sample. Our method is fast and scalable in that it significantly reduces the total computational cost as measured by the number of PDE solves, independent of the parameter dimension, the data dimension, and the number of greedy optimization iterations. The numerical experiments on both a linear advection-diffusion initial condition inversion problem and a nonlinear diffusion coefficient inversion problem illustrate the effectiveness and scalability of our method.

In this work we considered Gaussian prior and independent Gaussian additive noise, for which the Laplace approximation provides a good approximation of the posterior. In future work, we plan to develop (i) extension to nonlinear Bayesian OED problems with more general prior and noise distributions, (ii) other approximations of the posterior such as the student's t-distribution, and (iii) theoretical analysis of the convergence of the swapping greedy algorithm.

Appendix A. Proof of Theorem 3.3.

To start, we need the following results.

Proposition A.1 ([5]). *Let $\mathbf{A},\mathbf{B}\in\mathbb{C}^{n\times n}$ be Hermitian positive semidefinite with $\mathbf{A}\succeq\mathbf{B}$, and then*

$$0\le \log\det(\mathbf{I}+\mathbf{A})-\log\det(\mathbf{I}+\mathbf{B})\le \log\det(\mathbf{I}+\mathbf{A}-\mathbf{B}). \tag{A.1}$$

Proposition A.2 (Weinstein–Aronszajn identity [55]). *Let A and B be matrices of size $m\times n$ and $n\times m$, respectively, and then*

$$\det(\mathbf{I}_{n\times n}+\mathbf{B}\mathbf{A})=\det(\mathbf{I}_{m\times m}+\mathbf{A}\mathbf{B}). \tag{A.2}$$

From the definition of $\widetilde{\mathbf{H}}_m=\mathbf{\Gamma}_{\rm pr}^{\frac12}\mathbf{F}^T\mathbf{\Gamma}_{\rm n}^{-1}\mathbf{F}\mathbf{\Gamma}_{\rm pr}^{\frac12}$, we first denote $\hat{\mathbf{F}}=\mathbf{\Gamma}_{\rm n}^{-\frac12}\mathbf{F}$. Then

$$\hat{\mathbf{F}}=\mathbf{\Gamma}_{\rm n}^{-\frac12}\mathbf{W}\mathbf{F}_d=\mathbf{W}(\mathbf{\Gamma}_{\rm n}^{d})^{-\frac12}\mathbf{F}_d=\mathbf{W}\hat{\mathbf{F}}_d. \tag{A.3}$$

We can write the form of EIG in (3.32) as

$$\Psi(\mathbf{W})=\frac12\text{logdet}\left(\mathbf{I}_{n\times n}+\mathbf{\Gamma}_{\rm pr}^{\frac12}\mathbf{F}^T\mathbf{\Gamma}_{\rm n}^{-1}\mathbf{F}\mathbf{\Gamma}_{\rm pr}^{\frac12}\right) \tag{A.4a}$$

$$=\frac12\text{logdet}\left(\mathbf{I}_{n\times n}+\mathbf{\Gamma}_{\rm pr}^{\frac12}\hat{\mathbf{F}}^T\hat{\mathbf{F}}\mathbf{\Gamma}_{\rm pr}^{\frac12}\right) \tag{A.4b}$$

$$=\frac12\text{logdet}\left(\mathbf{I}_{n\times n}+\mathbf{\Gamma}_{\rm pr}^{\frac12}\hat{\mathbf{F}}_d^T\mathbf{W}^T\mathbf{W}\hat{\mathbf{F}}_d\mathbf{\Gamma}_{\rm pr}^{\frac12}\right). \tag{A.4c}$$

By the Weinstein–Aronszajn identity (A.2), we have

$$\Psi(\mathbf{W})=\frac12\text{logdet}\left(\mathbf{I}_{r\times r}+\mathbf{W}\hat{\mathbf{F}}_d\mathbf{\Gamma}_{\rm pr}^{\frac12}\mathbf{\Gamma}_{\rm pr}^{\frac12}\hat{\mathbf{F}}_d^T\mathbf{W}^T\right) \tag{A.5a}$$

$$=\frac12\text{logdet}\left(\mathbf{I}_{r\times r}+\mathbf{W}\mathbf{H}_d\mathbf{W}^T\right), \tag{A.5b}$$

where in the second equality we use the definition of $\mathbf{H}_d$. We denote the eigenvalue decomposition of $\mathbf{H}_d$ as

$$\mathbf{H}_d=\mathbf{U}_k\mathbf{\Sigma}_k\mathbf{U}_k^T+\mathbf{U}_\perp\mathbf{\Sigma}_\perp\mathbf{U}_\perp^T, \tag{A.6}$$

where $(\mathbf{\Sigma}_k, \mathbf{U}_k)$ represent the k dominant eigenpairs and $(\mathbf{\Sigma}_\perp, \mathbf{U}_\perp)$ represent the remaining $d-k$ eigenpairs. Then by the definition of the EIG approximation $\hat{\Psi}(\mathbf{W})$ in (3.34), we have

$$\begin{aligned}
&\text{(A.7a)} \quad \Psi(\mathbf{W}) - \hat{\Psi}(\mathbf{W}) = \frac{1}{2}\text{logdet}\left(\mathbf{I}_{r\times r} + \mathbf{W}\mathbf{H}_d\mathbf{W}^T\right) - \frac{1}{2}\text{logdet}\left(\mathbf{I}_{r\times r} + \mathbf{W}\hat{\mathbf{H}}_d\mathbf{W}^T\right)\\
&\text{(A.7b)} \quad \leq \frac{1}{2}\text{logdet}\left(\mathbf{I}_{r\times r} + \mathbf{W}\mathbf{H}_d\mathbf{W}^T - \mathbf{W}\hat{\mathbf{H}}_d\mathbf{W}^T\right)\\
&\text{(A.7c)} \quad = \frac{1}{2}\text{logdet}\left(\mathbf{I}_{r\times r} + \mathbf{W}\mathbf{U}_\perp\mathbf{\Sigma}_\perp\mathbf{U}_\perp^T\mathbf{W}^T\right)\\
&\text{(A.7d)} \quad = \frac{1}{2}\text{logdet}\left(\mathbf{I}_{r\times r} + \mathbf{W}\mathbf{U}_\perp\mathbf{\Sigma}_\perp^{1/2}\mathbf{\Sigma}_\perp^{1/2}\mathbf{U}_\perp^T\mathbf{W}^T\right)\\
&\text{(A.7e)} \quad = \frac{1}{2}\text{logdet}\left(\mathbf{I}_{(d-k)\times(d-k)} + \mathbf{\Sigma}_\perp^{1/2}\mathbf{U}_\perp^T\mathbf{W}^T\mathbf{W}\mathbf{U}_\perp\mathbf{\Sigma}_\perp^{1/2}\right),
\end{aligned}$$

where we use Proposition A.1 for the inequality and Proposition A.2 for the last equality. By definition of the design matrix $\mathbf{W}$ in (2.10), we have that $\mathbf{W}^T\mathbf{W} \in \mathbb{R}^{d\times d}$ is a square diagonal matrix with r diagonal entries as one and the others zero, which satisfies $\mathbf{W}^T\mathbf{W} \preceq \mathbf{I}_{d\times d}$. Consequently, we have

(A.8)
$$\Psi(\mathbf{W}) - \hat{\Psi}(\mathbf{W}) \leq \frac{1}{2}\text{logdet}\left(\mathbf{I}_{(d-k)\times(d-k)} + \mathbf{\Sigma}_\perp^{1/2}\mathbf{U}_\perp^T\mathbf{U}_\perp\mathbf{\Sigma}_\perp^{1/2}\right) = \frac{1}{2}\text{logdet}\left(\mathbf{I}_{(d-k)\times(d-k)} + \mathbf{\Sigma}_\perp\right),$$

where we use the orthonormality $\mathbf{U}_\perp^T\mathbf{U}_\perp = \mathbf{I}_{(d-k)\times(d-k)}$ for the eigenvectors in the equality. This concludes the upper bound for $\Psi(\mathbf{W}) - \hat{\Psi}(\mathbf{W})$. The lower bound in (3.35) is implied by Proposition A.1 and $\mathbf{H}_d \succeq \hat{\mathbf{H}}_d$.

REFERENCES

[1] A. Alexanderian, P. J. Gloor, and O. Ghattas, *On Bayesian A-and D-optimal experimental designs in infinite dimensions*, Bayesian Anal., 11 (2016), pp. 671–695, https://doi.org/10.1214/15-BA969.

[2] A. Alexanderian, N. Petra, G. Stadler, and O. Ghattas, *A-optimal design of experiments for infinite-dimensional Bayesian linear inverse problems with regularized ℓ_0-sparsification*, SIAM J. Sci. Comput., 36 (2014), pp. A2122–A2148, https://doi.org/10.1137/130933381.

[3] A. Alexanderian, N. Petra, G. Stadler, and O. Ghattas, *A fast and scalable method for A-optimal design of experiments for infinite-dimensional Bayesian nonlinear inverse problems*, SIAM J. Sci. Comput., 38 (2016), pp. A243–A272, https://doi.org/10.1137/140992564.

[4] A. Alexanderian, N. Petra, G. Stadler, and O. Ghattas, *Mean-variance risk-averse optimal control of systems governed by PDEs with random parameter fields using quadratic approximations*, SIAM/ASA J. Uncertain. Quantif., 5 (2017), pp. 1166–1192, https://doi.org/10.1137/16M106306X.

[5] A. Alexanderian and A. K. Saibaba, *Efficient D-optimal design of experiments for infinite-dimensional Bayesian linear inverse problems*, SIAM J. Sci. Comput., 40 (2018), pp. A2956–A2985, https://doi.org/10.1137/17M115712X.

[6] A. Alghamdi, M. Hesse, J. Chen, and O. Ghattas, *Bayesian poroelastic aquifer characterization from InSAR surface deformation data. Part* I*: Maximum a posteriori estimate*, Water Resour. Res., 56 (2020), e2020WR027391.

[7] A. Alghamdi, M. Hesse, J. Chen, U. Villa, and O. Ghattas, *Bayesian poroelastic aquifer characterization from InSAR surface deformation data. Part* II*: Quantifying the uncertainty*, Water Resour. Res., 57 (2021), e2021WR029775, https://doi.org/10.1029/2021WR029775.

[8] I. Ambartsumyan, W. Boukaram, T. Bui-Thanh, O. Ghattas, D. Keyes, G. Stadler, G. Turkiyyah, and S. Zampini, *Hierarchical matrix approximations of Hessians arising in inverse problems governed by PDEs*, SIAM J. Sci. Comput., 42 (2020), pp. A3397–A3426.

[9] N. Aretz-Nellesen, P. Chen, M. A. Grepl, and K. Veroy, *A-optimal experimental design for hyper-parameterized linear Bayesian inverse problems*, in Numerical Mathematics and Advanced Applications ENUMATH 2019, Lect. Notes Comput. Sci. Eng. 139, 2020, pp. 489–497.

[10] N. Aretz-Nellesen, M. A. Grepl, and K. Veroy, *3D-VAR for parameterized partial differential equations: A certified reduced basis approach*, Adv. Comput. Math., 45 (2019), pp. 2369–2400.

[11] A. C. Atkinson and A. N. Donev, *Optimum Experimental Designs*, Oxford University Press, Oxford, 1992.

[12] A. Attia, A. Alexanderian, and A. K. Saibaba, *Goal-oriented optimal design of experiments for large-scale Bayesian linear inverse problems*, Inverse Problems, 34 (2018), 095009.

[13] F. Bach, *Learning with submodular functions: A convex optimization perspective*, Found. Trends Mach. Learn., 6 (2013), pp. 145–373.

[14] J. Beck, B. M. Dia, L. F. Espath, Q. Long, and R. Tempone, *Fast Bayesian experimental design: Laplace-based importance sampling for the expected information gain*, Comput. Methods Appl. Mech. Engrg., 334 (2018), pp. 523–553, https://doi.org/10.1016/j.cma.2018.01.053.

[15] A. A. Bian, J. M. Buhmann, A. Krause, and S. Tschiatschek, *Guarantees for greedy maximization of non-submodular functions with applications*, in Proceedings of the 34th International Conference on Machine Learning, Proc. Mach. Learn. Res. 70, 2017, pp. 498–507.

[16] C. Boutsidis, M. W. Mahoney, and P. Drineas, *An Improved Approximation Algorithm for the Column Subset Selection Problem*, http://arxiv.org/abs/0812.4293, 2008.

[17] T. Bui-Thanh, C. Burstedde, O. Ghattas, J. Martin, G. Stadler, and L. C. Wilcox, *Extreme-scale UQ for Bayesian inverse problems governed by PDEs*, in Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, 2012.

[18] T. Bui-Thanh and O. Ghattas, *Analysis of the Hessian for inverse scattering problems. Part* I*: Inverse shape scattering of acoustic waves*, Inverse Problems, 28 (2012), 055001, https://doi.org/10.1088/0266-5611/28/5/055001.

[19] T. Bui-Thanh and O. Ghattas, *Analysis of the Hessian for inverse scattering problems. Part* II*: Inverse medium scattering of acoustic waves*, Inverse Problems, 28 (2012), 055002, https://doi.org/10.1088/0266-5611/28/5/055002.

[20] T. Bui-Thanh and O. Ghattas, *Analysis of the Hessian for inverse scattering problems. Part* III*: Inverse medium scattering of electromagnetic waves*, Inverse Probl. Imaging, 7 (2013), pp. 1139–1155.

[21] 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, https://doi.org/10.1137/12089586X.

[22] S. Chatterjee and A. S. Hadi, *Sensitivity Analysis in Linear Regression*, Wiley Ser. Probab. Stat. 327, John Wiley & Sons, New York, 2009.

[23] P. Chen and O. Ghattas, *Hessian-based sampling for high-dimensional model reduction*, Int. J. Uncertain. Quantif., 9 (2019).

[24] P. Chen and O. Ghattas, *Projected Stein variational gradient descent*, in Proceedings of Advances in Neural Information Processing Systems, 2020.

[25] P. Chen and O. Ghattas, *Taylor approximation for chance constrained optimization problems governed by partial differential equations with high-dimensional random parameters*, SIAM/ASA J. Uncertain. Quantif., 9 (2021), pp. 1381–1410.

[26] P. Chen, M. Haberman, and O. Ghattas, *Optimal design of acoustic metamaterial cloaks under uncertainty*, J. Comput. Phys., 431 (2021), 110114.

[27] P. Chen, U. Villa, and O. Ghattas, *Hessian-based adaptive sparse quadrature for infinite-dimensional Bayesian inverse problems*, Comput. Methods Appl. Mech. Engrg., 327 (2017), pp. 147–172, https://doi.org/10.1016/j.cma.2017.08.016.

[28] P. Chen, U. Villa, and O. Ghattas, *Taylor approximation and variance reduction for PDE-constrained optimal control under uncertainty*, J. Comput. Phys., 385 (2019), pp. 163–186; also available online from https://arxiv.org/abs/1804.04301.

[29] P. Chen, K. Wu, and O. Ghattas, *Bayesian inference of heterogeneous epidemic models: Application to COVID-19 spread accounting for long-term care facilities*, Comput. Methods Appl. Mech. Engrg., 385 (2021), 114020.

[30] B. Crestel, A. Alexanderian, G. Stadler, and O. Ghattas, *A-optimal encoding weights for nonlinear inverse problems, with application to the Helmholtz inverse problem*, Inverse Problems, 33 (2017), 074008.

[31] B. Crestel, G. Stadler, and O. Ghattas, *A comparative study of regularizations for joint inverse problems*, Inverse Problems, 35 (2018), 024003.

[32] Y. Daon and G. Stadler, *Mitigating the influence of boundary conditions on covariance operators derived from elliptic PDEs*, Inverse Probl. Imaging, 12 (2018), pp. 1083–1102; also available online from https://arxiv.org/abs/1610.05280.

[33] P. H. 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, https://doi.org/10.1137/090780717.

[34] S. Fujishige, *Submodular Functions and Optimization*, Ann. Discrete Math. 58, Elsevier, New York, 2005.

[35] N. Halko, P. G. Martinsson, and J. A. Tropp, *Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions*, SIAM Rev., 53 (2011), pp. 217–288.

[36] I. Han, P. Kambadur, K. Park, and J. Shin, *Faster greedy map inference for determinantal point processes*, in Proceedings of the 34th International Conference on Machine Learning, Proc. Mach. Learn. Res. 70, 2017, pp. 1384–1393.

[37] M. Hesse and G. Stadler, *Joint inversion in coupled quasistatic poroelasticity*, J. Geophys. Res. Solid Earth, 119 (2014), pp. 1425–1445.

[38] X. Huan and Y. M. Marzouk, *Simulation-based optimal Bayesian experimental design for nonlinear systems*, J. Comput. Phys., 232 (2013), pp. 288–317, https://doi.org/10.1016/j.jcp.2012.08.013.

[39] X. Huan and Y. M. Marzouk, *Gradient-based stochastic optimization methods in Bayesian experimental design*, Int. J. Uncertain. Quantif., 4 (2014), pp. 479–510.

[40] X. Huan and Y. M. Marzouk, *Sequential Bayesian Optimal Experimental Design via Approximate Dynamic Programming*, preprint, arXiv:1604.08320, 2016.

[41] T. Isaac, N. Petra, G. Stadler, and O. Ghattas, *Scalable and efficient algorithms for the propagation of uncertainty from data through inference to prediction for large-scale problems, with application to flow of the Antarctic ice sheet*, J. Comput. Phys., 296 (2015), pp. 348–368, https://doi.org/10.1016/j.jcp.2015.04.047.

[42] J. Jagalur-Mohan and Y. Marzouk, *Batch Greedy Maximization of Non-Submodular Functions: Guarantees and Applications to Experimental Design*, preprint, arXiv:2006.04554, 2020.

[43] A. G. Kalmikov and P. Heimbach, *A Hessian-based method for uncertainty quantification in global ocean state estimation*, SIAM J. Sci. Comput., 36 (2014), pp. S267–S295.

[44] M. R. Khodja, M. D. Prange, and H. A. Djikpesse, *Guided Bayesian Optimal Experimental Design*, Inverse Problems, 26 (2010), 055008, https://doi.org/10.1088/0266-5611/26/5/055008.

[45] A. Krause and D. Golovin, *Submodular function maximization*, in Tractability: Practical Approaches to Hard Problems, Cambridge University Press, Cambridge, 2014, pp. 71–104.

[46] Q. Long, M. Motamed, and R. Tempone, *Fast Bayesian optimal experimental design for seismic source inversion*, Comput. Methods Appl. Mech. Engrg., 291 (2015), pp. 123–145, https://doi.org/10.1016/j.cma.2015.03.021.

[47] Q. Long, M. Scavino, R. Tempone, and S. Wang, *Fast estimation of expected information gains for Bayesian experimental designs based on Laplace approximations*, Comput Methods Appl Mech. Engrg., 259 (2013), pp. 24–39.

[48] K. Manohar, B. W. Brunton, J. N. Kutz, and S. L. Brunton, *Data-driven sparse sensor placement for reconstruction: Demonstrating the benefits of exploiting known patterns*, IEEE Control Syst., 38 (2018), pp. 63–86, https://doi.org/10.1109/MCS.2018.2810460.

[49] J. Martin, L. C. Wilcox, C. Burstedde, and O. Ghattas, *A stochastic Newton MCMC method for large-scale statistical inverse problems with application to seismic inversion*, SIAM J. Sci. Comput., 34 (2012), pp. A1460–A1487, https://doi.org/10.1137/110845598.

[50] G. Nemhauser, L. Wolsey, and M. Fisher, *An analysis of approximations for maximizing submodular set functions*—I, Math. Program., 14 (1978), pp. 265–294, https://doi.org/10.1007/BF01588971.

[51] T. O'Leary-Roseberry, U. Villa, P. Chen, and O. Ghattas, *Derivative-informed projected neural networks for high-dimensional parametric maps governed by PDEs*, Comput. Methods Appl. Mech. Engrg., 388 (2022), 114199.

[52] D. Papailiopoulos, A. Kyrillidis, and C. Boutsidis, *Provable deterministic leverage score sampling*, in Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2014, pp. 997–1006.

[53] N. Petra, J. Martin, G. Stadler, and O. Ghattas, *A computational framework for infinite-dimensional Bayesian inverse problems: Part* II. *Stochastic Newton MCMC with application to ice sheet flow inverse problems*, SIAM J. Sci. Comput., 36 (2014), pp. A1525–A1555.

[54] N. Petra, H. Zhu, G. Stadler, T. J. R. Hughes, and O. Ghattas, *An inexact Gauss-Newton method for inversion of basal sliding and rheology parameters in a nonlinear Stokes ice sheet model*, J. Glaciol., 58 (2012), pp. 889–903, https://doi.org/10.3189/2012JoG11J182.

[55] C. Pozrikidis, *An Introduction to Grids, Graphs, and Networks*, Oxford University Press, Oxford, 2014.

[56] F. Pukelsheim, *Optimal Design of Experiments*, SIAM, Philadelphia, 2006.

[57] A. K. Saibaba, A. Alexanderian, and I. C. Ipsen, *Randomized matrix-free trace and log-determinant estimators*, Numer. Math., 137 (2017), pp. 353–395.

[58] A. K. Saibaba and P. K. Kitanidis, *Fast computation of uncertainty quantification measures in the geostatistical approach to solve inverse problems*, Adv. Water Resour., 82 (2015), pp. 124–138.

[59] A. M. Stuart, *Inverse problems: A Bayesian perspective*, Acta Numer., 19 (2010), pp. 451–559, https://doi.org/10.1017/S0962492910000061.

[60] S. Subramanian, K. Scheufele, M. Mehl, and G. Biros, *Where did the tumor start? An inverse solver with sparse localization for tumor growth models*, Inverse Problems, 36 (2020), 045006, https://doi.org/10.1088/1361-6420/ab649c.

[61] U. Villa, N. Petra, and O. Ghattas, *HIPPYlib: An extensible software framework for large-scale inverse problems governed by PDEs: Part* I*: Deterministic inversion and linearized Bayesian inference*, ACM Trans. Math. Software, 47 (2021), https://doi.org/10.1145/3428447.

[62] K. Wei, R. Iyer, and J. Bilmes, *Submodularity in data subset selection and active learning*, in Proceedings of the 32nd International Conference on Machine Learning, Lille, France, Proc. Mach. Learn. Res., F. Bach and D. Blei, eds., 2015, pp. 1954–1963.

[63] J. Worthen, G. Stadler, N. Petra, M. Gurnis, and O. Ghattas, *Towards adjoint-based inversion for rheological parameters in nonlinear viscous mantle flow*, Phys. Earth Planet Inter., 234 (2014), pp. 23–34, https://doi.org/10.1016/j.pepi.2014.06.006.

[64] K. Wu, P. Chen, and O. Ghattas, *An efficient method for goal-oriented linear Bayesian optimal experimental design: Application to optimal sensor placement*, SIAM/ASA J. Uncertain. Quantif., to appear; also available online from, preprint, arXiv:2102.06627, 2022.

[65] K. Wu, T. O'Leary-Roseberry, P. Chen, and O. Ghattas, *Large-scale Bayesian optimal experimental design with derivative-informed projected neural network*, J. Sci. Comput., to appear; also available online from, preprint, arXiv:2201.07925, 2022.

[66] S. Yang, G. Stadler, R. Moser, and O. Ghattas, *A shape Hessian-based boundary roughness analysis of Navier–Stokes flow*, SIAM J. Appl. Math., 71 (2011), pp. 333–355, https://doi.org/10.1137/100796789.

[67] J. Yu, V. M. Zavala, and M. Anitescu, *A scalable design of experiments framework for optimal sensor placement*, J. Process Control, 67 (2018), pp. 44–55.

[68] H. Zhu, S. Li, S. Fomel, G. Stadler, and O. Ghattas, *A Bayesian approach to estimate uncertainty for full waveform inversion with a priori information from depth migration*, Geophysics, 81 (2016), pp. R307–R323.

[69] H. Zhu, N. Petra, G. Stadler, T. Isaac, T. J. R. Hughes, and O. Ghattas, *Inversion of geothermal heat flux in a thermomechanically coupled nonlinear stokes ice sheet model*, The Cryosphere, 10 (2016), pp. 1477–1494.