COMPUTATIONAL ASPECTS OF OPTIMAL EXPERIMENTAL DESIGN FOR LARGE-SCALE BAYESIAN LINEAR INVERSE PROBLEMS

NICK NEUBERGER*, BART G VAN BLOEMEN WAANDERS†, AND ALEN ALEXANDERIAN ‡

Abstract. The advent of high-performance-computing and scalable methods for tackling complex PDE-driven problems has opened the door to the use of powerful tools such as Bayesian inversion and optimal experimental design (OED) on a new scale. This document presents several applications of OED tools within the context of an example model: the steady-state convection-diffusion PDE. Estimating the source term of this equation is an example of a linear inverse problem. We first focus on deterministic inversion with the introduction of fundamental concepts and operators needed later in Bayesian inversion. Bayesian inversion is then used to estimate the source in a probabilistic sense. Solving the Bayesian problem motivates the OED formulation. In this context, a sensor configuration is sought such that uncertainty in the estimated parameter is minimized. Specifically, we focus on computing A-optimal sensor placements, where we seek to minimize average posterior variance. Our emphasis is in computational aspects of OED for infinite-dimensional inverse problems. We conclude our computational experiments with a demonstration of the effectiveness of optimal design strategy. Specifically, we demonstrate that the A-optimal design outperforms naive or random sensor placements.

1. Introduction.

We consider optimal experimental design (OED) for infinite-dimensional Bayesian linear inverse problems governed by partial differential equations (PDEs). Specifically, we focus on optimal placement of sensors, where measurement data are collected. Solving such problems requires an intricate combination of methods and theories from probability, linear algebra, PDEs, and mathematical and numerical analysis. Also, in practice, one has to work within a computational budget. In this work, we explore computational aspects of OED for large-scale Bayesian linear inverse problem with an eye toward implementations on HPC architectures.

1.1. The model problem.

We focus on a stationary convection-diffusion model for example dynamics. It is sufficiently complex to produce some interesting results, yet simple enough to facilitate the omission of unnecessary problem-dependent matters. Take $\Omega \subset \mathbb{R}^2$ to be a bounded domain and consider the following stationary PDE:

$$\begin{aligned} -\alpha \Delta u + \boldsymbol{\nu} \cdot \nabla u = m(\boldsymbol{x}) \quad & \text{in } \Omega, \\ u \equiv 0 \quad & \text{on } \Gamma_1, \\ \nabla u \cdot \boldsymbol{n} \equiv 0 \quad & \text{on } \Gamma_2. \end{aligned} \tag{1.1}$$

Here, u is the state variable. We assume that $\Omega = (0,1)^2$ and that both the diffusion α and the advection field $\boldsymbol{\nu}$ are constant. Additionally, the composite boundary is such that $\Gamma_1 \cup \Gamma_2 = \partial\Omega$, with $\boldsymbol{n}$ being the outward normal unit vector to the boundary. We let the source $m \in L^2(\Omega)$ be the inversion parameter that we seek to estimate.

One can formulate an inverse problem in the deterministic and Bayesian settings. The deterministic formulation seeks to find a representation of the inversion parameter that reproduces the data most accurately under the governing model. On the other hand, the Bayesian approach is used to construct a statistical description of the parameter that accounts for uncertainties in parameter estimation. Realizations of the Bayesian posterior law may be propagated through the model resulting in a probability distribution on the state variable. In both approaches, we must contend with noisy data and the ill-posedness of the problem, due to sparse measurements and smoothing properties of the forward mapping.

*Department of Mathematics, North Carolina State University, jnneuber@ncsu.edu

†Center for Computing Research, Sandia National Labs, bartv@sandia.gov

‡Department of Mathematics, North Carolina State University, alexanderian@ncsu.edu

In what follows, we discuss both deterministic and Bayesian formulations of the inverse problem, before tackling the OED problem of determining an optimal sensor placement. We will see that the OED problem is built on the Bayesian inverse problem. Thus, the understanding and manipulation of the components of the Bayesian formulation is required before we can develop the OED problem formulation and its solution methods. However, before discussing the matter of uncertainty in parameter estimation, it is instructive to consider the deterministic setting. This will allow us to build the key operators and components required in subsequent problems. Once this is done, converting the deterministic approach into the Bayesian formulation is straightforward, in the case of linear inverse problems.

1.2. Related work. The field of Bayesian inverse problems and optimal experimental design contain many open areas of research. A major motivation for the development of these fields are the numerous practical engineering and scientific applications. The formulation and solution of such problems requires an assortment of theoretical and computational tools at one's disposal. This is the reason why research pertaining to Bayesian inverse problems and OED necessitates rigorous knowledge of underlying theory and efficient implementation of algorithms. While the present work focuses on linear Bayesian inverse problems and A-optimal designs, there exists extensive works on design of nonlinear Bayesian inverse problem; see e.g., [9, 5, 1, 12].

The works [4, 2, 3], which concern OED for Bayesian linear inverse problems are closely related to the present work. In particular, [2] provides an accessible introduction on finite dimensional Gaussian random variables, linear inverse problems in the finite dimensional setting, and *Bayes Risk*, which we discuss later. The paper [3] provides a detailed perspective on OED for Bayesian linear inverse problems in Hilbert space with focus on Bayesian A- and D-optimal designs. The article [4] details a scalable approach for tackling the linear OED problem in a manner suitable for HPC architectures.

1.3. Finite element method and the observation operator. We discretize (1.1) in the space defined by $\mathbb{V}_n := \operatorname{span}\{\phi_i(\boldsymbol{x})\}_{i=1}^n$, which is the space spanned by the finite-element (FE) basis $\{\phi_i\}_{i=1}^n$. For all computations, we choose ϕ_i to be Lagrange nodal-basis-functions. For $v \in L^2(\Omega)$, FE coefficients can be obtained such that

$$v \approx v_h := \sum_{i=1}^n v_i \phi_i.$$

In particular, the discretized inversion parameter is given by $m_h = \sum_i m_i \phi_i$. In a Bayesian formulation, we use measurement data in conjunction with prior knowledge about the inversion parameter to estimate the vector $\boldsymbol{m} = [m_1\ m_2\ \cdots\ m_n]^\top$ of FE coefficients of m_h.

Let $\boldsymbol{d} \in \mathbb{R}^d$ denote the vector of measurement data, collected at locations $\{\boldsymbol{x}_j\}_{j=1}^d$, where $\boldsymbol{x}_j \in \Omega$. Data are often collected at uniform or random sensor locations. This is sufficient for demonstrating the mechanics of the inverse problem and solution methods, but may be suboptimal in applications. We discuss this later when formulating the optimal experimental design problem, which concerns placing sensors in “optimal” locations.

We now introduce the first of several operators that appear in formulation of the inverse problem. Let $\mathcal{B} : L^2(\Omega) \to \mathbb{R}^d$ be the observation map. This operator takes a function in $L^2(\Omega)$ and produces measurements at the locations described by $\{\boldsymbol{x}_j\}$. Note that the observation map does not account for noise when collecting data.

2. Deterministic inversion. Much of the problem structure developed in the deterministic formulation will translate to the Bayesian setting. For this reason, various constructs are carefully built in the following section. In a deterministic formulation, we formulate the inverse problem as that of finding an m that minimizes

$$\mathcal{J}(m) := \frac{1}{2}\|\mathcal{B}u - \boldsymbol{d}\|_2^2 + \mathcal{R}(m;\gamma), \tag{2.1}$$

where u is obtained by solving (1.1). The first term of the functional is referred to as the data-misfit. This term is a measure of the error between model observations and data $\boldsymbol{d}$. It is assumed that $\boldsymbol{d} = \mathcal{B}u + \boldsymbol{\eta}$, where the additive noise is normally distributed with mean zero and covariance $\sigma^2\mathbf{I}$. That is, $\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$. Note that this assumes uncorrelated measurements. The second term of $\mathcal{J}$ is a regularization term. For illustration in this section, we consider an L^2-regularization, $\mathcal{R}(m;\gamma) := \frac{\gamma}{2}\|m\|^2_{L^2(\Omega)}$. This term is required to manage the ill-posedness of the problem. The deterministic inverse problem maybe solved numerically in one of the following ways:

1. Use variational methods to form a system that results in $m^* \in L^2(\Omega)$, the minimizer of $\mathcal{J}$. Then, discretize this system to obtain the FE coefficients $\boldsymbol{m}^*$, which uniquely characterize the approximation m_h^* of m^*.
2. Discretize (1.1) and (2.1), and then use variational methods to obtain a system that uniquely determines $\boldsymbol{m}^*$, the FE coefficients of m_n^*.

Approach 1 is referred to as the optimize-then-discretize method (OTD), and 2 as discretize-then-optimize (DTO) approach. Due to the linearity of the present inverse problem, both OTD and DTO methods result in the symmetric positive definite system:

$$(\mathbf{F}^*\mathbf{F} + \gamma\mathbf{I})\,\boldsymbol{m} = \mathbf{F}^*\boldsymbol{d}. \tag{2.2}$$

As discussed below, $\mathbf{F}$ is the discretized parameter-to-observable map and $\mathbf{F}^*$ is its adjoint.

2.1. Discretization issues. To define the adjoint operator $\mathbf{F}^*$ in (2.2), we need to pay close attention to the inner product on the discretized inversion parameter space. This discretized space is $\mathbb{R}^n$ equipped with the following *mass-weighted* inner product:

$$(\boldsymbol{u}, \boldsymbol{v})_{\mathbf{M}} = \boldsymbol{v}^T\mathbf{M}\boldsymbol{u},$$

where $\mathbf{M}$ is the finite element mass matrix, $M_{ij} = \int_\Omega \phi_i\phi_j$ with $i, j \in \{1, \dots, n\}$. Note that this mass weighted inner product is none but the discretized $L^2(\Omega)$ inner product. To see this, let $(\cdot,\cdot)$ be the $L^2(\Omega)$ inner product and let u_h and v_h be in $\mathbb{V}_n$. Then,

$$(u_h, v_h) = \left(\sum_{j=1}^n u_j\phi_j, \sum_{i=1}^n v_i\phi_i\right) = \sum_{i,j=1}^n v_i u_j(\phi_i, \phi_j) = \boldsymbol{v}^T\mathbf{M}\boldsymbol{u} = (\boldsymbol{u}, \boldsymbol{v})_{\mathbf{M}}.$$

Note that $\boldsymbol{u}$ and $\boldsymbol{v}$ are the vectors of FE coefficients corresponding to u_h and v_h, respectively. In what follows, we let $\mathbb{R}^n_{\mathbf{M}}$ denote the inner product space $\mathbb{R}^n$ equipped with the mass-weighted inner product.

Next, we describe the remaining components of (2.2), the matrices $\mathbf{F}$ and $\mathbf{F}^*$. The map $\mathbf{F} : \mathbb{R}^n_{\mathbf{M}} \to \mathbb{R}^d$ is a discretization of the parameter-to-observable map $\mathcal{F} : L^2(\Omega) \to \mathbb{R}^d$ and $\mathbf{F}^* : \mathbb{R}^d \to \mathbb{R}^n_{\mathbf{M}}$ is a discretization of the adjoint map $\mathcal{F}^* : \mathbb{R}^d \to L^2(\Omega)$. Additionally, denote $\bar{\mathbf{F}} : \mathbb{R}^n_{\mathbf{M}} \to \mathbb{R}^n_{\mathbf{M}}$ as the discretization of the parameter-to-state map $\bar{\mathcal{F}} : L^2(\Omega) \to H^1(\Omega)$. Obtaining these matrices (or their applications) is discussed shortly below. The last operator we discretize is the observation map. Denote $\mathbf{B}$ as the discretization of the observation operator $\mathcal{B}$. It is important to note that constructing this operator is not only problem-dependent, but also dependent upon the selected software. A reasonable approach to extract a measurement value from a function $v \in \mathbb{V}_n$ at a point $\boldsymbol{x} \in \Omega$ is to numerically integrate v against a sufficiently scaled Gaussian-like function that approximates the δ distribution centered at $\boldsymbol{x}$. Doing so is equivalent to obtaining a local average of v about the point $\boldsymbol{x}$. This process is linear and a single matrix $\mathbf{B} : \mathbb{R}^n_{\mathbf{M}} \to \mathbb{R}^d$ can perform simultaneous observations, approximating the action of $\mathcal{B}$.

Following a DTO strategy, the discretized objective functional with L^2-regularization is

$$\mathcal{J}(\boldsymbol{m}) := \frac{1}{2}\|\mathbf{B}\boldsymbol{u} - \boldsymbol{d}\|_2^2 + \frac{\gamma}{2}\boldsymbol{m}^T\mathbf{M}\boldsymbol{m}, \tag{2.3}$$

where $\boldsymbol{u}$ solves the discretized state equation, which we discuss next. Define the matrix $A_{ij} := \alpha(\nabla\phi_i, \nabla\phi_j) + (\phi_i, \boldsymbol{\nu}\cdot\nabla\phi_j)$. The discretized state equation is described as follows:

$$\boldsymbol{v}^T\mathbf{A}\boldsymbol{u} = \boldsymbol{v}^T\mathbf{M}\boldsymbol{m},\ \forall\boldsymbol{v} \in \mathbb{R}^n_{\mathbf{M}} \iff \mathbf{A}\boldsymbol{u} = \mathbf{M}\boldsymbol{m}. \tag{2.4}$$

This system gives an explicit representation of the discretized parameter-to-state and parameter-to-observable maps:

$$\begin{array}{ll} \text{parameter-to-state:} & \bar{\mathbf{F}} := \mathbf{A}^{-1}\mathbf{M}, \\ \text{parameter-to-observable:} & \mathbf{F} := \mathbf{B}\bar{\mathbf{F}}. \end{array}$$

In practice, an application of $\mathbf{F}$ (a state solve) requires solving a linear system. Using $\mathbf{F}$, $\mathcal{J}$ can be redefined in terms of $\boldsymbol{m}$:

$$\mathcal{J}(\boldsymbol{m}) := \frac{1}{2}\|\mathbf{F}\boldsymbol{m} - \boldsymbol{d}\|_2^2 + \frac{\gamma}{2}\boldsymbol{m}^T\mathbf{M}\boldsymbol{m}. \tag{2.5}$$

It can be shown that $\nabla\mathcal{J}(\boldsymbol{m}) = 0$ results in (2.2).

2.2. Computational results. Computing the minimizer of $\mathcal{J}$ requires implementing routines that perform applications of the forward and adjoint operators, $\mathbf{F}$ and $\mathbf{F}^*$ and then solving a symmetric positive definite (SPD) system using a Krylov iterative method. This process is matrix free. It does not require building the forward and adjoint operators, which is an important consideration in large-scale problems. The expressions that define the forward/adjoint equations can be assembled using FE software. We use the FEinCS 2019 library [6] in Python for discretization, PDE solves, and assembly of relevant matrices. In the case of the steady-state convection/diffusion model, applications of the forward/adjoint operators require solving linear systems.

Performing deterministic inversion is equivalent to minimizing (2.3) over $\mathbb{R}^n_{\mathbf{M}}$. Once the FE coefficients that minimize $\mathcal{J}$ are obtained, the expansion of the minimizer over $\mathbb{V}_n$ may be plotted and relative error calculated. Hence, the true parameter may be compared with its approximation. We implement an equally spaced FE mesh of size $n = n_{\boldsymbol{x}}^2$ with k^2 uniformly distributed interior sensors. The sub-boundary Γ_2 is the $x = 1$ side of the unit square. Taking $\boldsymbol{\nu} = [1.5, 0]$ provides "flow" out of the boundary. In the present study, the true parameter is given by $m_{\text{true}}(\boldsymbol{x}) = f(\boldsymbol{x}) + g(\boldsymbol{x})$, where

$$f(\boldsymbol{x}) = \begin{cases} 1, & \boldsymbol{x} \in, R \\ 0, & \boldsymbol{x} \notin R \end{cases} \qquad \text{and} \quad g(\boldsymbol{x}) = 1.5\left\{-\frac{1}{2(0.15)^2}\left[(x_1 - 0.5)^2 + (x_2 - 0.5)^2\right]\right\},$$

and R is the rectangle $R(\boldsymbol{x}) = [0.125, 0.5] \times [0.375, 0.625]$. Physical and problem-specific parameters are provided in the Table 2.1. Note that 15^2 uniformly distributed sensors are selected. Results are obtained with synthetic data such that $\|\boldsymbol{d} - \mathbf{B}\boldsymbol{u}\|_2/\|\boldsymbol{d}\|_2 = 0.0016$. This fraction is viewed as the relative magnitude of the noise.

Parameter	$n_{\boldsymbol{x}}$	k	α	γ
Value	20	15	0.06	0.007

Table 2.1: Parameters for the deterministic inversion problem.

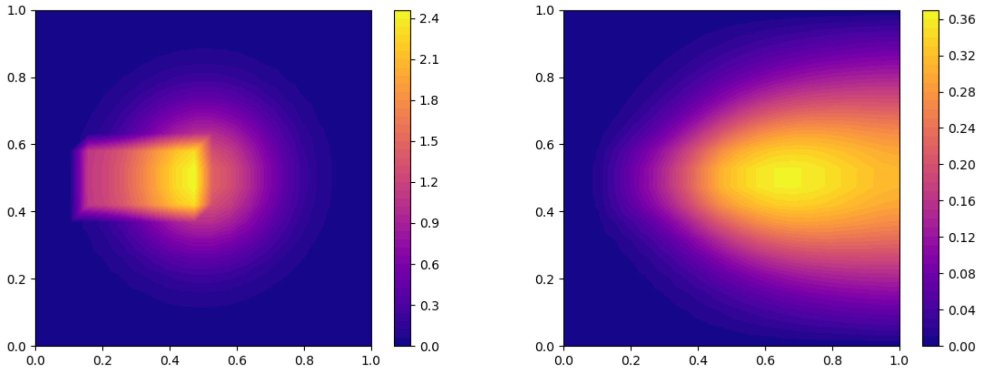

Fig. 2.1: The true parameter m_{true} used to generate data (left) and the corresponding state solution u (right).

We see that the true parameter has been advected in the positive x-direction, resulting in outflow. In addition to this, the amplitude of the parameter is decreased by the diffusive properties of the model. Performing the inversion, we see the estimate captures the features of the true parameter relatively well. Quantitatively, a mass weighted relative error of 0.0790 is achieved. We use the mass weighted inner product to measure the relative error since we are concerned about the error in approximating the true solution in the space $\mathbb{V}_n$. The true parameter has sharp edges, which are notoriously difficult to reconstruct in diffusion models. Nonetheless, all major features in m_{true} are present in the reconstruction.

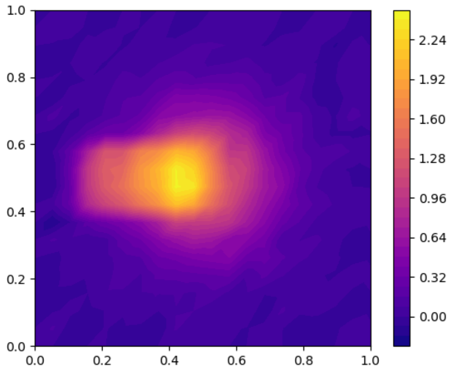

Fig. 2.2: Solution to the deterministic inverse problem.

3. Bayesian inversion. Here, we discuss the Bayesian formulation of the inverse problem under study. The objective of the Bayesian inverse problem is to obtain a probabilistic description of the inversion parameter from noisy data. The present discussion is based on the developments in [7], which outlines a computational framework for solving infinite-dimensional Bayesian linear inverse problems. For theory of Bayesian inverse problems in the infinite-dimensional setting, the readers can refer to [10].

Recall the infinite-dimensional representation of the parameter-to-observable map $\mathcal{F} : L^2(\Omega) \to \mathbb{R}^d$. It is assumed that noise is additive and Gaussian with mean zero and covariance denoted by $\mathbf{\Gamma}_{\text{noise}}$:

$$\boldsymbol{d} = \mathcal{F}m + \boldsymbol{\eta}, \;\; \boldsymbol{\eta} \sim \mathcal{N}(0, \mathbf{\Gamma}_{\text{noise}}). \tag{3.1}$$

Note that for the present example, the parameter-to-observable map is a continuous linear transformation. The Bayesian formulation entails forming a prior law—a prior statistical description of the inversion parameter informed by whatever information is available. The prior, in conjunction with the data-likelihood, the law of the data conditioned on the inversion parameter, facilitates the use of Bayes' Theorem. Assuming a Gaussian prior, application of Bayes' theorem implies that the posterior law, for the present inverse problem, is Gaussian with analytic expressions for the posterior mean and covariance. In this case, the posterior mean coincides with the *maximum a posteriori probability (MAP) point.* This and the posterior covariance are sufficient in describing the posterior law. Before we provide the formulas for the MAP point and the posterior covariance, we discuss the definition of a suitable Gaussian prior in the present infinite-dimensional setting.

3.1. The prior. The prior mean, given by $m_0 \in L^2(\Omega)$, may be informed by problem-specific information. When no such information is available, the prior mean may be set as the zero function. This is our decision for all subsequent computations. A clever choice of prior covariance operator is the square of the solution operator of an elliptic PDE. This provides advantageous smoothing properties and the important property of being *trace-class*, positive, and selfadjoint. Additionally, the prior covariance being the *square* of the inverse of the elliptic operator makes applications of the square root of the covariance straightforward [7]. Let $s \in L^2(\Omega)$ and $a_1, a_2 > 0$. Then, the mentioned elliptic PDE is

$$\begin{aligned} -a_1 \Delta m + a_2 m &= s, \quad \text{in } \Omega, \\ \nabla m \cdot \boldsymbol{n} &= 0, \quad \text{on } \partial\Omega. \end{aligned}$$

This corresponds to the weak form

$$a_1(\nabla m, \nabla p) + a_2(m, p) = (s, p), \;\; \forall p \in H^1(\Omega). \tag{3.2}$$

If the operator $\mathcal{A}$ is such that $\mathcal{A}m = s$, then the solution operator of the PDE is $\mathcal{A}^{-1}$. The prior covariance is defined to be the square of this: $\mathcal{C}_0 = \mathcal{A}^{-2}$. It is also important to note the role of the parameters a_1 and a_2, responsible for the behavior of the prior law. Here, the prior law describes a Gaussian random field, whose correlation length and pointwise variance are controlled by a_1 and a_2, respectively.

For computational purposes, we need the FE coordinates of the prior mean, $\boldsymbol{m}_0$, and the discretized prior covariance, $\mathbf{\Gamma}_{\text{pr}}$. To arrive at the prior covariance, we discretize (3.2) with regards to the FE basis. Define the matrix $\mathbf{K}$ according to $K_{ij} = (\nabla\phi_i, \nabla\phi_j)$. Then, the discretized elliptic equation is given by

$$\mathbf{M}^{-1}(a_1\mathbf{K} + a_2\mathbf{M})\boldsymbol{m} = \boldsymbol{s}.$$

Additionally, define $\mathbf{E} := \mathbf{M}^{-1}(a_1\mathbf{K} + a_2\mathbf{M})$, then it can be shown that $\mathbf{E}$ and $\mathbf{E}^{-1}$ are selfadjoint with respect to the mass-weighted inner product $(\cdot,\cdot)_{\mathbf{M}}$. Using the FE coefficients of the prior mean and the discretized elliptic operator, we introduce the prior distribution:

$$\pi_{\text{pr}}(\boldsymbol{m}) \propto \exp\left\{-\frac{1}{2}\|\mathbf{E}(\boldsymbol{m} - \boldsymbol{m}_0)\|_{\mathbf{M}}^2\right\}. \tag{3.3}$$

This also indicates $\boldsymbol{\Gamma}_{\text{pr}} := \mathbf{E}^{-2}$.

3.2. The likelihood. The likelihood law provides a probabilistic description of the data conditioned on the inversion parameter. Since the noise is Gaussian with mean zero, the likelihood is also Gaussian:

$$\pi_{\text{like}}(\boldsymbol{d}|\boldsymbol{m}) \propto \exp\left\{-\frac{1}{2}\|\mathbf{F}\boldsymbol{m} - \boldsymbol{d}\|^2_{\boldsymbol{\Gamma}^{-1}_{\text{noise}}}\right\}. \tag{3.4}$$

The fact that $\boldsymbol{d}|\boldsymbol{m} \sim \mathcal{N}(\mathbf{F}\boldsymbol{m}, \boldsymbol{\Gamma}_{\text{noise}})$ follows from the independence of $\boldsymbol{m}$ and $\boldsymbol{\eta}$. We also note the role of the noise covariance by viewing the formula above. In the case of uncorrelated measurements, or when $\boldsymbol{\Gamma}_{\text{noise}} = \sigma^2\mathbf{I}_d$, sensor locations corresponding to high-noise measurements will not be influential. This is due to the fact the diagonal entry of $\boldsymbol{\Gamma}^{-1}_{\text{noise}}$ corresponding to the high-noise measurement will effectively be zero. We will return to this idea later in the OED discussion.

3.3. The posterior. Since the prior and likelihood are given in discretized form, Bayes' Theorem can be applied as follows:

$$\pi_{\text{post}}(\boldsymbol{m}) \propto \pi_{\text{like}}(\boldsymbol{d}|\boldsymbol{m})\pi_{\text{pr}}(\boldsymbol{m}). \tag{3.5}$$

By substituting the expressions for the likelihood and prior density in the above formula and some algebraic manipulations, we can describe the posterior law explicitly: the posterior law is a Gaussian with mean $\boldsymbol{m}_{\text{map}}$ and covariance $\boldsymbol{\Gamma}_{\text{post}}$ defined according to

$$\boldsymbol{m}_{\text{map}} = \boldsymbol{\Gamma}_{\text{post}}\left(\mathbf{F}^*\boldsymbol{\Gamma}^{-1}_{\text{noise}}\boldsymbol{d} + \boldsymbol{\Gamma}^{-1}_{\text{pr}}\boldsymbol{m}_0\right), \quad \boldsymbol{\Gamma}_{\text{post}} = \left(\mathbf{F}^*\boldsymbol{\Gamma}^{-1}_{\text{noise}}\mathbf{F} + \boldsymbol{\Gamma}^{-1}_{\text{pr}}\right)^{-1}. \tag{3.6}$$

It is important to note several details about the posterior information:

- Applications of $\boldsymbol{\Gamma}_{\text{post}}$ require a linear solve.
- Obtaining $\boldsymbol{m}_{\text{post}}$ requires a linear solve.
- Defining $\boldsymbol{\Gamma}_{\text{noise}} = \sigma^2\mathbf{I}_d$ simplifies calculations.

3.4. Computational considerations. Being able to solve the Bayesian inversion problem is essential to constructing the OED problem. For this reason, it is prudent to perform various theoretical verifications such as verifying the implementation of the adjoint operator and the gradient and Hessian expressions. To make calculations as fast as possible, it is necessary to use suitable matrix factorizations and matrix-free approaches when available. Doing so vastly decreases computational time and results in an approach suitable for large scale problems (it may be practical to impose some computational budget).

To illustrate some of the computational issues, we consider the computation of the MAP point. This is done by solving the system

$$\boldsymbol{\Gamma}^{-1}_{\text{post}}\boldsymbol{m} = \mathbf{F}^*\boldsymbol{\Gamma}^{-1}_{\text{noise}}\boldsymbol{d} + \boldsymbol{\Gamma}^{-1}_{\text{pr}}\boldsymbol{m}_0$$

using a Krylov iterative approach. To enable this, we create a routine that applies $\boldsymbol{\Gamma}^{-1}_{\text{post}}$ to a vector. An application of $\boldsymbol{\Gamma}^{-1}_{\text{post}}$ requires both a state and adjoint solve, as well as a multiplication of $\boldsymbol{\Gamma}^{-1}_{\text{pr}}$ with a vector. Matrix-free Krylov iterative methods are essential for obtaining the MAP point quickly. Due to SPD system structure, we rely on the (preconditioned) Conjugate Gradient (CG) for solving the resulting linear system. For more details on computational aspects of solving Bayesian linear inverse problems, see [7].

3.5. Computational results. We perform an illustrative computational study in the Bayesian framework as we did in the deterministic setting. The same discretization method, boundary conditions, advection, and true parameter are used. Doing so will allow for the comparison between the minimizer obtained in the deterministic formulation and the Bayesian MAP point. A significant difference between the two problems, related to our discussions of OED, is that measures for the quality of the reconstructions maybe obtained naturally in the Bayesian setting. Namely, we may consider various ways of characterizing the uncertainty in the estimated parameters. For example, the trace of the posterior covariance quantifies the average variance of the posterior distribution. We can also study the percentage of average variance reduction from performing inversion. This is represented by the quantity $\Delta\text{Var} := 100 \cdot (1 - \text{tr}[\mathbf{\Gamma}_{\text{post}}]/\text{tr}[\mathbf{\Gamma}_{\text{prior}}])$. We note that for large-scale problems, computing the trace directly is inefficient. For this reason, a randomized trace estimator is used. This will be discussed further in Section 4.

In the present example, we use a Gaussian prior as described in Section 3.1. Regarding the noise model, we assume an additive Gaussian noise model, and let the noise covariance be of the form $\mathbf{\Gamma}_{\text{noise}} = \sigma^2 \mathbf{I}_d$. We summarize the parameters defining the various aspects of the Bayesian inverse problem under study in Table 3.1.

Parameter	$n_{\boldsymbol{x}}$	k	α	a_1	a_2	σ^2
Value	20	15	0.06	0.01	0.08	0.001

Table 3.1: Parameters for the Bayesian inversion problem.

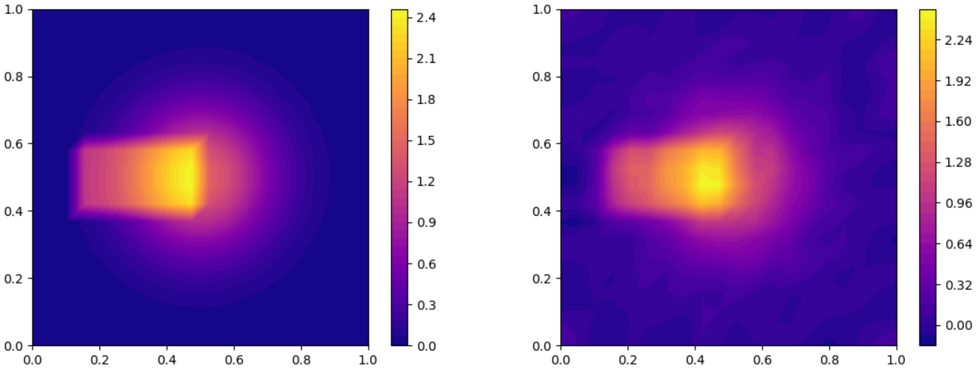

Fig. 3.1: The true parameter m_{true} used to generate data (left) and m_{map} (right).

Results are obtained with synthetic noise level is set such that $\|\boldsymbol{d} - \mathbf{B}\boldsymbol{u}\|_2 / \|\boldsymbol{d}\|_2 = 0.0021$. Qualitatively, the MAP point captures the major features of m_{true}. The relative error between the true parameter and MAP point is calculated to be 0.0777. Note that this is in par with our results from deterministic inversion. Regarding uncertainty reduction, we observed an average variance reduction of $\Delta\text{Var} = 99.8692\%$. A variance reduction of this magnitude should not be expected in real applications. The reasons for such a large uncertainty reduction can be attributed to a “wide prior” and dense sensor network. By a "wide prior" we mean that the parameters governing the prior (a_1 and a_2) are selected such that the prior has a large average variance.

4. Optimal experimental design.

The purpose of solving the OED problem is to obtain optimal sensor locations in which to collect data. In this context, a design is the placement of measurement locations within the domain Ω and optimality is determined through a measure of uncertainty associated with the inversion parameters. Our overall strategy aligns closely to that in [4]. Specifically, we consider Bayesian A-optimal designs that seek to minimize the average posterior variance.

Let $\mathbb{S} := \{\boldsymbol{x}_j\}_{j=1}^{N_s},\ \boldsymbol{x}_i \in \Omega$, be the set of possible sensor locations, otherwise known as the candidate sensor locations. To each location $\boldsymbol{x}_j$, we assign a weight, w_j. If $\boldsymbol{w} = [w_1, \cdots, w_{N_s}]^T$ is the vector containing the design weights, the optimal design is obtained by minimizing the OED functional

$$\Psi(\boldsymbol{w}) := \Theta(\boldsymbol{w}) + \Phi(\boldsymbol{w};\beta),\ \boldsymbol{w} \in [0,1]^{N_s}. \tag{4.1}$$

The functional $\Theta : \mathbb{R}^{N_s} \to \mathbb{R}$ is referred to as the *design criterion.* Deciding upon a criterion is typically problem-dependent. Specifically, the choice of criterion depends upon problem structure, the chosen definition of uncertainty, and related quantities of interest. The functional $\Phi : \mathbb{R}^{N_s} \times \mathbb{R}_+ \to \mathbb{R}$ is the *penalty* function whose role is to promote sparsity. Increasing the penalty parameter β results in sparser designs.

4.1. The design criterion.

To arrive at a design criterion, the design weights must be introduced into the Bayesian inversion problem. This is done by defining a suitably weighted data-likelihood. Let $\mathbf{W} \in \mathbb{R}^{N_s \times N_s}$ be a diagonal matrix such that $\text{diag}\,[\mathbf{W}] = \boldsymbol{w}$ and $\boldsymbol{\Gamma}_{\text{noise}} = \sigma^2 \mathbf{I}_{N_s}$ for all subsequent calculations. Then,

$$\pi_{\text{like}}(\boldsymbol{d}|\boldsymbol{m};\boldsymbol{w}) \propto \exp\left\{-\frac{1}{2}\|\mathbf{W}^{1/2}(\mathbf{F}\boldsymbol{m} - \boldsymbol{d})\|^2_{\boldsymbol{\Gamma}^{-1}_{\text{noise}}}\right\}. \tag{4.2}$$

Note that in the case when $\mathbf{W} = \mathbf{I}_{N_s}$, the original likelihood (3.4) is obtained. In the context of binary weights, this is equivalent to collecting data at all possible measurement locations. In the case of nonbinary weights, we view (4.2) as placing bias upon specific sensor locations. Candidate locations $\boldsymbol{x}_j$ corresponding to larger w_j indicate these locations are highly influential in terms of reducing uncertainty. The corresponding "weighted" posterior covariance operator is given by

$$\boldsymbol{\Gamma}_{\text{post}}(\boldsymbol{w}) = \left(\sigma^{-2}\mathbf{F}^*\mathbf{W}\mathbf{F} + \boldsymbol{\Gamma}^{-1}_{\text{pr}}\right)^{-1}. \tag{4.3}$$

It is worth noting that the data $\boldsymbol{d}$ does not appear in the posterior covariance. In the present study, we focus on the Bayesian A-optimality criterion:

$$\Theta(\boldsymbol{w}) := \text{tr}[\boldsymbol{\Gamma}_{\text{post}}(\boldsymbol{w})]. \tag{4.4}$$

For the present linear Gaussian problem, this A-optimality criterion is equal to the Bayes Risk, which is an average mean square error of the posterior mean; see e.g., [2]. Thus, under our provided assumptions, reducing the average posterior variance corresponds to reducing parameter MAP estimation inaccuracy in an average sense. This will be revisited in our computational studies later in this section.

4.2. Penalty of the OED objective.

As discussed, the role of the penalty term is to enforce sparsity in sensor placement. For simplicity, we consider an ℓ_1-norm penalty:

$$\Phi(\boldsymbol{w}) := \beta\|\boldsymbol{w}\|_1. \tag{4.5}$$

This choice of Φ is convex. However, w_j may take on any value in $[0,1]$. This means that the $\boldsymbol{w}^*$ that minimizes Ψ is not a binary vector. Our aim is to modulate β such that a desirable number of weights are nearly zero. Sensor locations corresponding to the remaining weights comprise the (approximate) optimal design. Since $\boldsymbol{w} \in [0,1]^{N_s}$, the gradient is computed as $\nabla\Phi(\boldsymbol{w}) = \beta\mathbb{1}$. This will be necessary for utilizing a gradient-based optimization routine with box constraints in order to solve the OED problem. Although our choice of penalty function is sufficient for demonstration and application of the OED problem, more sophisticated penalty approaches exists. For example, in [4] a sequence of penalty functions is used to approximate the ℓ_0-norm.

4.3. Evaluating the criterion and its gradient. As seen in the previous section, evaluating the penalty and its gradient for $\boldsymbol{w} \in \mathbb{R}^{N_s}$ is straightforward. It is not the same case for the criterion. We outline a matrix-free approach, which is based on the developments in [4]. We begin by recalling that for a given weight vector $\boldsymbol{w}$, obtaining the MAP point is equivalent to minimizing

$$\hat{\mathcal{J}}(\boldsymbol{m};\boldsymbol{w}) = \frac{1}{2\sigma^2}\|\mathbf{W}^{1/2}(\mathbf{F}\boldsymbol{m}-\boldsymbol{d})\|^2 + \frac{1}{2}\|\boldsymbol{\Gamma}_{\text{pr}}^{-1/2}(\boldsymbol{m}-\boldsymbol{m}_0)\|^2_{\mathbf{M}}. \tag{4.6}$$

A simple calculation show that the Hessian of $\hat{\mathcal{J}}$ is

$$\mathbf{H}(\boldsymbol{w}) = \frac{1}{\sigma^2}\mathbf{F}^*\mathbf{W}\mathbf{F} + \boldsymbol{\Gamma}_{\text{pr}}^{-1}.$$

Defining the Hessian of the data-misfit term as $\mathbf{H}_{\text{misfit}}(\boldsymbol{w}) := \frac{1}{\sigma^2}\mathbf{F}^*\mathbf{W}\mathbf{F}$ we write,

$$\mathbf{H}(\boldsymbol{w}) = \mathbf{H}_{\text{misfit}}(\boldsymbol{w}) + \boldsymbol{\Gamma}_{\text{pr}}^{-1}.$$

Thus, the posterior covariance can be defined in terms of the Hessian as

$$\boldsymbol{\Gamma}_{\text{post}}(w) = \mathbf{H}^{-1}(\boldsymbol{w}).$$

Building the posterior covariance explicitly in order to compute the trace is not scalable. We will see shortly that having access to applications of $\mathbf{H}^{-1}(\boldsymbol{w})$ is sufficient for estimating $\Theta(\boldsymbol{w}) = \operatorname{tr}[\boldsymbol{\Gamma}_{\text{post}}(\boldsymbol{w})]$ and its gradient. With Monte-Carlo methods, one can estimate the trace of an SPD matrix $\mathbf{A}$:

$$\operatorname{tr}[\mathbf{A}] \approx \frac{1}{N_{tr}}\sum_{i=1}^{N_{tr}} (\boldsymbol{y}_i, \mathbf{A}\boldsymbol{y}_i)_2\,,$$

where $\boldsymbol{y}_i$ are independent random vectors. It is not as simple as applying this concept to $\Theta(\boldsymbol{w})$, given that an evaluation of Θ at the point $\boldsymbol{w}$ should approximate the trace of the infinite-dimensional representation of $\boldsymbol{\Gamma}_{\text{post}}$. For this reason, it is necessary to appropriately scale the random vectors $\boldsymbol{y}_i$. With this idea, we redefine the criterion in terms of its trace estimation:

$$\Theta(\boldsymbol{w}) := \frac{1}{N_{tr}}\sum_{i=1}^{N_{tr}} \left(\boldsymbol{z}_i, \mathbf{H}^{-1}\boldsymbol{z}_i\right)_{\mathbf{M}}\,, \text{ for } \boldsymbol{z}_i := \mathbf{M}^{-1/2}\boldsymbol{y}_i. \tag{4.7}$$

See [4] for a derivation of the mass-weighted trace estimator.

Next, the gradient can be derived. The following equations and assumptions will be used:

1. $\frac{\partial \mathbf{H}^{-1}(\boldsymbol{w})}{\partial w_j} = -\mathbf{H}^{-1}(\boldsymbol{w})\frac{\partial \mathbf{H}(\boldsymbol{w})}{w_j}\mathbf{H}^{-1}(\boldsymbol{w})$

2. $\mathbf{H}^{-1}(\boldsymbol{w})$ is $\mathbf{M}$-symmetric
3. $\frac{\partial \mathbf{H}_{\text{misfit}}(\boldsymbol{w})}{w_j} = \sigma^{-2}\mathbf{F}^* \boldsymbol{e}_j \boldsymbol{e}_j^T \mathbf{F}$, where $\boldsymbol{e}_j$ is the j^{th} standard unit vector in $\mathbb{R}^{N_s}$.

With the above assumptions and defining $\boldsymbol{q}_i = \mathbf{H}^{-1}\boldsymbol{z}_i$, we have that

$$\begin{aligned} \frac{\partial \Theta(\boldsymbol{w})}{\partial w_j} &= \frac{1}{N_{tr}} \frac{\partial}{\partial w_j} \sum_{i=1}^{N_{tr}} \left(\boldsymbol{z}_i, \mathbf{H}^{-1}(\boldsymbol{w})\boldsymbol{z}_i\right)_{\mathbf{M}} \\ &= -\frac{1}{N_{tr}} \sum_{i=1}^{N_{tr}} \left(\boldsymbol{z}_i, \mathbf{H}^{-1}(\boldsymbol{w}) \frac{\partial \mathbf{H}_{\text{misfit}}(\boldsymbol{w})}{\partial w_j} \mathbf{H}^{-1}(\boldsymbol{w})\boldsymbol{z}_i\right)_{\mathbf{M}} \\ &= -\frac{1}{N_{tr}} \sum_{i=1}^{N_{tr}} \left(\boldsymbol{q}_i, \frac{\partial \mathbf{H}_{\text{misfit}}(\boldsymbol{w})}{\partial w_j} \boldsymbol{q}_i\right)_{\mathbf{M}}. \end{aligned}$$

Note that $\frac{\partial \mathbf{H}(\boldsymbol{w})}{\partial w_j} = \frac{\partial \mathbf{H}_{\text{misfit}}(\boldsymbol{w})}{\partial w_j}$ was used. Denote $\boldsymbol{d}_i = \mathbf{F}\boldsymbol{q}_i$ then

$$\left(\boldsymbol{q}_i, \frac{\partial \mathbf{H}_{\text{misfit}}(\boldsymbol{w})}{\partial w_j} \boldsymbol{q}_i\right)_{\mathbf{M}} = \frac{1}{\sigma^2}\left(\boldsymbol{q}_i, \mathbf{F}^* \boldsymbol{e}_j \boldsymbol{e}_j^T \mathbf{F}\boldsymbol{q}_i\right)_{\mathbf{M}} = \frac{1}{\sigma^2}\left(\mathbf{F}\boldsymbol{q}_i, \boldsymbol{e}_j \boldsymbol{e}_j^T \mathbf{F}\boldsymbol{q}_i\right)_2 = \frac{1}{\sigma^2}\left[\boldsymbol{d}_i^{(j)}\right]^2.$$

Finally, the gradient of Θ can be defined:

$$\frac{\partial \Theta(\boldsymbol{w})}{w_j} = -\frac{1}{\sigma^2 N_{tr}} \sum_{i=1}^{N_{tr}} \left[\boldsymbol{d}_i^{(j)}\right]^2, \; j = 1, \cdots N_s. \tag{4.8}$$

4.4. Computational considerations. The previous section provides enough information to implement a gradient-based optimization algorithm for the purpose of minimizing $\Psi(\boldsymbol{w})$. However, a practical implementation for large-scale problems is rather involved. See [4] for details. The implementations in the present study are simplistic and are intended for illustration. First, $\mathbf{F}$, $\mathbf{F}^*$, and $\boldsymbol{\Gamma}_{pr}^{-1}$ are explicitly constructed. (In practical implementations, we need to implement routines that provide applications of the forward/adjoint operators.) A gradient-based optimization routine is then selected. In our implementations, we use the limited memory Broyden–Fletcher–Goldfarb–Shanno with bound constraints (*L-BFGS-B*) algorithm as implemented in the SciPy library [11] in Python. The algorithm was originally introduced in [13].

For a given $\boldsymbol{w}$, $\boldsymbol{q}_i$ are computed via SPD solves. In optimizing $\Psi(\boldsymbol{w}) = \Theta(\boldsymbol{w}) + \Phi(\boldsymbol{w}; \beta)$, evaluations of Θ and its gradient are computed as described in the previous section.

Due to our choice of the penalty approach, the computed optimal design will, in general, be a nonbinary vector. However, the ℓ_1-penalty ensures that some of the weights are nearly zero. In our numerical studies, we consider these weights as zero and select sensors corresponding to larger weights.

4.5. Computational results. In application, the number of sensors desired is likely to be a small number. The intuition behind this is as follows: if the number of sensors is such that the physical domain is saturated, moving the sensors around is unlikely to make a notable difference. On the other hand, when we have access to only a few sensors, their placement is crucial. For this reason, in our computational studies we seek Designs that are comprised of relatively few sensors.

Our results were obtained with a FE-mesh of size $n = n_{\boldsymbol{x}}^2$. The Neumann boundary Γ_2 represents the $y = 0$ and $x = 1$ sides of the domain, and the vector-field $\boldsymbol{\nu} = [1.5, -1.5]$ produces outflow in the Southeast direction. Data is generated such that $\|\boldsymbol{d} - \mathbf{B}\boldsymbol{u}\|_2 / \|\boldsymbol{d}\|_2 =$

0.006. Lastly, the sparsity parameter β was selected to produce relatively few sensors and the true parameter is given by

$$m_{\text{true}}(\boldsymbol{x}) = 8 \exp\left\{ -\frac{1}{2(0.17)^2} \left[(x_1 - 0.5)^2 + (x_2 - 0.5)^2 \right] \right\}.$$

Parameter	$n_{\boldsymbol{x}}$	k	α	a_1	a_2	σ^2	β
Value	20	18	0.06	0.2905	0.151	0.1	0.2

Table 4.1: Parameters for the OED problem.

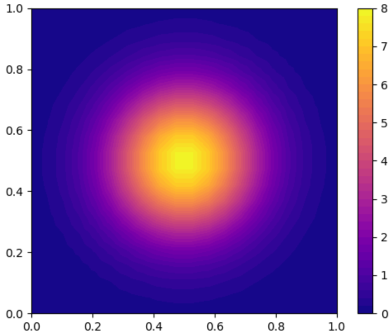

Fig. 4.1: The true source used to generate data: m_{true}.

The parameters in Table 4.1 result in an optimal design consisting of 9 sensor locations. We compare the OED result to results obtained using 9 uniform sensors located on the interior of Ω, and 9 randomly selected locations. The average variance reduction of the OED, uniform, and random solutions are 86.6488%, 78.4059%, and 73.0988% respectively.

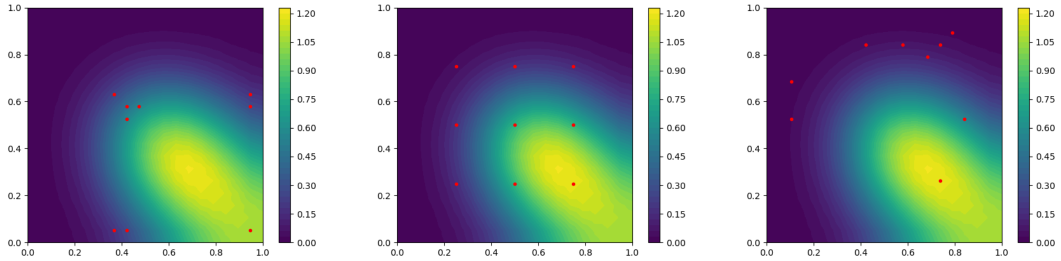

Fig. 4.2: The optimal design consisting of 9 sensors is depicted (left), a design of 9 uniform sensors (middle), and a design randomly distributed design (right). The designs are plotted over the state solution generated by the true parameter, m_{true}.

It is important to highlight that all observed random placements were sub-optimal. Designs are plotted over the state solution. In this context, the optimal design is not surprising. We would not expect to see measurement locations in portions of the region devoid of dynamics, such as the Northwest corner of the plots. This is why the optimal placement is symmetric about the outflow direction. It is noteworthy that the random sensors performed poorly due to placement of sensors in uninformative locations and that the uniform sensors were much better than random (in most cases).

Next, we study the results of sensor placement as the number of desired sensors changes. As mentioned before, optimal sensor placement is critical when only a small number of sensors are used. On the other hand, as the number of sensors grow, even a random placement of sensor maybe be effective. This is studied in Figure 4.3. In that figure, each point corresponds to a design consisting of k randomly placed sensors. This random design is used to generate synthetic data for the Bayesian inverse problem using a fixed m_{true}. We then compute the relative error between the true parameter and MAP point and the average posterior variance (i.e., the A-optimal criterion). The relative error is then plotted against the A-optimal criterion. For each value of k, 15 different randomly generated designs are considered, resulting in clusters of size 15 for each value of k. As we increase k, the variance of the clusters decreases and the clusters tend towards the origin. This indicates that random designs with a large number of sensors behave similarly and are generally effective in reducing posterior uncertainty and producing a good quality MAP point.

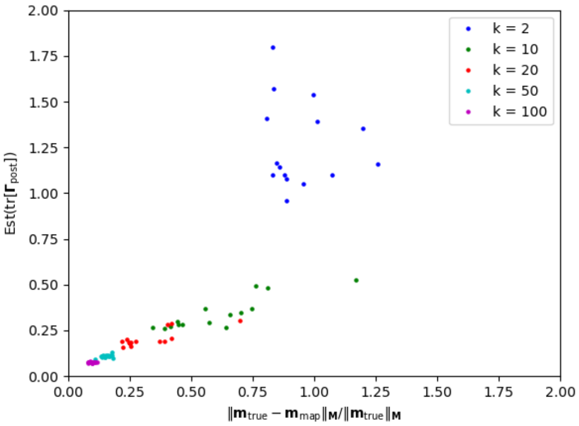

Fig. 4.3: Five clusters corresponding to different numbers of sensors. For example, each dot in the $k = 2$ cluster corresponds to a random design consisting of 2 sensors. Using a fixed m_{true}, synthetic data of length 2 is generated and the Bayesian inverse problem is solved. The mass weighted relative error between the MAP point and true parameter are then plotted against the posterior average variance.

The trend of the clusters in Figure 4.3 is generally linear. This is not a coincidence, but a consequence of the relationship between Bayes Risk and the posterior covariance. Take data $\boldsymbol{y}$ and a Gaussian prior with law $\mu_{\text{pr}} = \mathcal{N}(\boldsymbol{m}_{\text{pr}}, \mathbf{\Gamma}_{\text{pr}})$. It can be shown that under the additional assumptions of a linear model and additive-Gaussian noise, the average mean square error between $\boldsymbol{m}$ and $\boldsymbol{m}^{\boldsymbol{y}}_{\text{map}}$ (Bayes Risk), is equal to the trace of the posterior covariance [8, 2]. Specifically, if we denote Bayes Risk as $\overline{\text{Risk}}(\boldsymbol{m}^{\boldsymbol{y}}_{\text{map}})$, then

$$\overline{\text{Risk}}(\boldsymbol{m}^{\boldsymbol{y}}_{\text{map}}) := \mathbb{E}_{\mu_{\text{pr}}}\mathbb{E}_{\mu_{\text{like}}}\left\{\|\boldsymbol{m} - \boldsymbol{m}^{\boldsymbol{y}}_{\text{map}}\|_2^2\right\}$$

$$= \int\int \|\boldsymbol{m} - \boldsymbol{m}^{\boldsymbol{y}}_{\text{map}}\|_2^2 \,\pi_{\text{like}}(\boldsymbol{y}|\boldsymbol{m})d\boldsymbol{y}\,\mu_{\text{pr}}(d\boldsymbol{m}) = \text{tr}[\boldsymbol{\Gamma}_{\text{post}}]. \quad (4.9)$$

We have simulated this theoretical expectation in a simple inverse problem governed by a one-dimensional heat equation; see Figure 4.4.

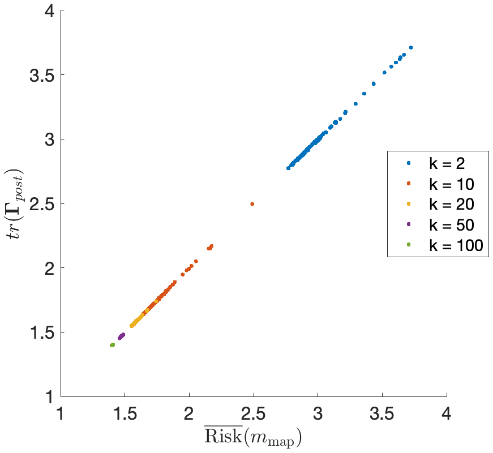

Fig. 4.4: Five clusters corresponding to different numbers of sensors.

In Figure 4.4, we see the theoretically expected result that the trace of the posterior covariance operator equals Bayes Risk. In this study, Bayes risk is computed via sampling: prior samples are used to generate noisy data required to solve the inverse problem, and the error in the MAP point is averaged over these samples. We see that the spread of the clusters corresponding to random designs of size k decrease as k grows. As in Figure 4.3, the clusters also tend towards the origin. This is yet another demonstration of the fact that when we have a large number of sensors, optimizing their placement is less consequential.

5. Conclusions.

In our brief study we explored mathematical and computational aspects of solving Bayesian linear inverse problems and optimal sensor placement for these class of problems. To make the presentation accessible, we presented concepts in increasing level of complexity, starting from deterministic inversion and building up to formulation of an OED problem. The driving application for our studies was volume source inversion in an advection diffusion equation in a two-dimensional geometry. A key motivation for the present study was to demarcate computational techniques that allow for OED problem in the linear setting to be solved in a scalable way in an HPC setting.

Our most insightful computational results were in the case of the OED problem. We compared our optimal design against uniform and random designs of the same size, observing which design was most successful in reduction of average variance. It was seen that the optimal design performed better than the uniform design, which performed better than the random design. Obtaining optimal designs in the context of our problem lead to further discussion on the relationship between MAP estimation error and average variance reduction, in terms of the number of sensors used. We numerically demonstrated that as the size of the design increases, the MAP point estimation error and average variance also decrease. This is intuitive and indicates that a practitioner may not need to solve an OED problem if sensor budget is not an issue. On the other hand, if only a few sensors are available, then optimal sensor placement is crucial. Additionally, we numerically demonstrated a theoretical result regarding equivalence of the standard A-optimality criterion and the Bayes Risk.

REFERENCES

[1] A. Alexanderian, *Optimal experimental design for infinite-dimensional Bayesian inverse problems governed by PDEs: a review*, Inverse Problems, 37 (2021), p. 043001.

[2] A. Alexanderian, *On Bayes risk of the posterior mean in linear inverse problems*, 2023.

[3] A. Alexanderian, P. J. Gloor, and O. Ghattas, *On Bayesian A- and D-Optimal Experimental Designs in Infinite Dimensions*, Bayesian Analysis, 11 (2016), pp. 671 – 695.

[4] 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 Journal on Scientific Computing, 36 (2014), pp. A2122–A2148.

[5] ———, *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.

[6] M. S. Alnaes et al., *The fenics project version 1.5*, Archive of Numerical Software, 3 (2015).

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

[8] K. Chaloner and I. Verdinelli, *Bayesian experimental design: A review*, Statistical science, (1995), pp. 273–304.

[9] E. Haber, L. Horesh, and L. Tenorio, *Numerical methods for the design of large-scale nonlinear discrete ill-posed inverse problems*, Inverse Problems, 26 (2010), p. 025002.

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

[11] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, *SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python*, Nature Methods, 17 (2020), pp. 261–272.

[12] K. Wu, P. Chen, and O. Ghattas, *A fast and scalable computational framework for large-scale high-dimensional Bayesian optimal experimental design*, SIAM/ASA Journal on Uncertainty Quantification, 11 (2023), pp. 235–261.

[13] C. Zhu, R. H. Byrd, P. Lu, and J. Nocedal, *Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization*, ACM Trans. Math. Softw., 23 (1997), p. 550–560.