Optimal experimental design for Bayesian inverse problems

Karina Koval

August 6, 2018

1 Introduction

Many problems in science and engineering require one to infer or reconstruct some parameter of interest from indirect observations and a mathematical model (forward operator) relating parameters to observables. Examples where inverse problem theories are used include image reconstruction, wave-based material inversion, weather predictions, geophysics, and petroleum engineering. Having access to informative data is integral to a good reconstruction of the parameters. In the worst case, the cost of data collection or physical restrictions often limit how much data one can collect. In the best case, even if we have access to large stores of data, processing all of it can be expensive and it can be full of redundancies, as poor experimental design choice can limit how much we can learn about our parameters. A natural question to ask then, is how can we design experimental conditions for data collection to optimally reconstruct/infer parameters of interest. This is called optimal experimental design (OED).

Before we can tackle an OED problem, we will first provide a brief background covering all the necessary pre-requisites for the OED problem. This will involve a mathematical description of the prototypical inverse problem. The standard method to solve an inverse problem is called the "deterministic approach". With this approach, solving the inverse problem is equivalent to solving a minimization problem to obtain a candidate which most likely gave rise to the data. However, many inverse problems are ill-posed. In particular, many parameters may be consistent with the data. The deterministic approach gives us one candidate for the solution, but it says nothing about how "likely" that candidate is versus other parameters. Thus, it is often preferable to have a complete statistical description of the parameters that are consistent with our data. Obtaining this statistical description is done by solving the inverse problem via a Bayesian approach.

In the Bayesian framework, we integrate any prior knowledge we have and combine it with our observations and model to obtain a probability distribution for the parameters[1]. This probability distribution allows us to determine the most likely candidate (maximum a posteriori or MAP point), but it has the additional benefit of providing us with some idea of how certain we are this was the true parameter (this being one aspect of what we call uncertainty quantification). We will see that

[1]Note that there is no inherent uncertainty in the parameter... there is a true parameter which gave rise to the data. We use probability as a tool to express/model our lack of perfect knowledge.

this Bayesian approach yields a natural description of what we mean by "optimal" reconstruction of parameters. the goals and the focus of the paper. However, I'm wondering if its voice could be made more authoritative and its goal could be made a bit more focused. (For instance, in the first sentence, I think "will give an overview of recent work that attempts to answer this optimal experimental design question" sounds a bit tentative; furthermore, from what I understand, the bulk of the paper seems to be an introduction to inverse problems, Bayesian inversion, and what OED is; the recent work portion seems to be given as an example/extension rather than the focus of the paper?) This paper will give an overview of recent work that answers a particular OED question [1]. Specifically, following the paper, we will focus on the optimal design of experiments for Bayesian inverse problems with forward operators that involve the solution of a partial differential equation (PDE). In section 2, we introduce a running example of a Bayesian inverse problem. This example serves to provide intuition for the abstract concepts discussed in this paper. Additionally, we use the example to motivate OED for a specific class of inverse problems. Section 3 touches on key ideas and difficulties from the field of inverse problems necessary to understand the remainder of the paper. More specifically, 3.1 introduces the prototypical inverse problem and explains why it is "hard". 3.2 formulates the Bayesian approach to solving inverse problems in a finite-dimensional setting, and 3.2.1 restricts this approach to work for problems with a similar structure to our motivating example given in 2. Section 4 gives a brief overview of the general optimal experimental design problem and discusses a specific formulation keeping the motivating example in mind. A further research direction is briefly touched upon in section 6.

2 Motivating example

In the event of a chemical attack in New York City, one can ask where we should place chemical sensors to be able to pinpoint the initial release of the chemical and most efficiently track the trajectory to make quick evacuation decisions.

We can formulate a simplified version of this problem as an optimal experimental design problem. For a simple version of this problem, we consider New York to be a 2D domain, Ω, with a few rectangular buildings, Γ_i. The transport of the chemical could be modeled with an advection-diffusion PDE (1) where the diffusion is driven by the wind field in NYC ($\boldsymbol{v}$).

The forward model maps an initial condition, $u_0 \in \mathcal{L}^2(\Omega)$ to the state space through the solution $u(\boldsymbol{x}, t)$ of the following PDE:

$$\begin{aligned} u_t - \kappa\Delta u + \boldsymbol{v}\cdot\nabla u = 0 \quad & \text{in } \Omega \times (0,T) \\ u(\cdot,0) = u_0 \quad & \text{in } \Omega \\ \kappa\nabla u\cdot\boldsymbol{n} = 0 \quad & \text{in } \partial\Omega \times (0,T) \end{aligned} \tag{1}$$

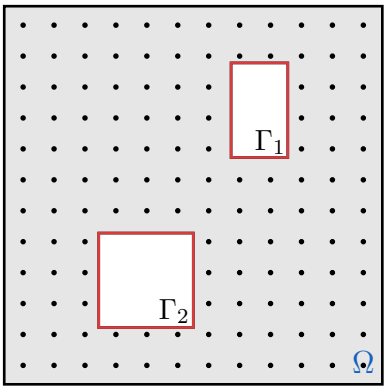


Figure 1: The domain (Ω) and possible sensor locations (depicted by black dots in the figure) for the 2D advection-diffusion problem. Here, Γ_1, Γ_2 denote the boundaries of two buildings.

Given any initial condition, u_0, we can simulate the chemical diffusion. The problem is then formulated as follows: given a grid of N_s possible locations for sensor placement and some knowledge of typical wind fields, what subset of sensors should we choose to optimally infer the initial condition (the location/concentration of the initial chemical attack) while keeping our monetary expenditures as low as possible. The next few sections will serve to provide the necessary background to be able to tackle problems such as this.

3 Background

In the previous section, our end goal is to reconstruct the initial condition (u_0) given observed data taken at a subset of sensor locations and knowledge of the underlying physical process governing the diffusion of the chemical. This is an example of an inverse problem. In this section, we give a general definition of inverse problems and discuss one method used to solve them. 3.1 and 3.2 will give abstract definitions and formulations while 3.2.1 will make these concepts concrete by referring back to our motivating example in section 2.

3.1 Inverse Problems

We start with the following relation:

$$\boldsymbol{d} = F(p) + \eta \tag{2}$$

Here, p is a parameter or parameter field we wish to reconstruct, it could be a vector in $\mathbb{R}^n$ (for example, if one is trying to reconstruct an image), or a function in some space $\mathcal{H}$ (this is the case in 2, our parameter is the initial condition). Our data is given by $\boldsymbol{d}$, and it is often a vector, say $\boldsymbol{d} \in \mathbb{R}^m$ (in 2 the data corresponds to point-wise measurements of the chemical concentration at the sensors). In most applications, the data is corrupted by noise. All sources of error, be it measurement or model error, are lumped into the variable η (for this paper we will assume that the error is additive), and we assume that the error is characterized by a distribution. F is the forward model and it maps the parameter space to the observation space. In many inverse problems governed by physical phenomena, F involves the solution

of a PDE (in 2, the forward model involves the solution of the advection-diffusion equation to obtain the chemical concentration in the domain as well as the restriction of the solution to point-wise measurements at the sensor locations).

In an inverse problem, one is given a model, F, as well as observations of the state or output $\boldsymbol{d}$, and attempts to solve for p. Can we be guaranteed that a solution exists, and if so, can we expect it to be unique? To answer this question, we recall the definition of well-posedness given by Jacques Hadamard. A problem is well-posed if a solution exists, is unique, and depends continuously on the data.

If at least one of the conditions is not met, then the problem is said to be ill-posed. It turns out that many inverse problems are ill-posed. Let $F \in \mathbb{R}^{m\times n}$, take $\boldsymbol{X} = \mathbb{R}^n$ and assume that $\text{rank}(F) = \min(m, n)$. It is easy to see how the first condition could be violated if the dimension of our observations is larger than the dimension of our parameters, i.e., we have an overdetermined system and we can not guarantee the existence of a solution. On the contrary, if our data lives in a smaller dimensional space than our parameter, we have an under-determined system and many parameters may be consistent with our data (analogous arguments hold for infinite dimensional parameters). The advection-diffusion inversion problem presented in section 2 is a good example where the third property is violated. The diffusive part kills off high-frequency information in the initial condition, so there could be arbitrarily large perturbations in the initial condition which have no effect on the data.

3.2 Bayesian inversion

The method of Bayesian inversion is one way to deal with the ill-posedness of inverse problems. In the Bayesian approach, probability is used as a tool to incorporate our uncertainty in the solution. Rather than finding the "most likely" parameter which gave rise to our data (this is referred to as the deterministic approach), the solution of the Bayesian inverse problem is a probability density for our parameter. Not only does this density give us automatic access to this "most likely" parameter, it also gives us a way to quantify how much we have learned and in special cases how much we can possibly learn[2]. Sampling this density allows us to obtain multiple candidates for the "true" parameter.

It is necessary to note that many inverse problems are infinite dimensional, i.e., you are trying to find a function in some Hilbert space which gave rise to finite dimensional data through a solution of a PDE. Often, it is instructive to formulate the method of Bayesian inversion in Hilbert spaces, however, since the goal of this paper is to introduce the concept behind this approach, we focus on Bayesian inversion in a finite dimensional setting. The ideas presented here carry over nicely to an infinite dimensional framework, but extra care must be taken to make them mathematically rigorous. For a rigorous infinite dimensional formulation, we refer interested readers to [4].

For Bayesian inversion, we treat p, $\boldsymbol{d}$ and η as random variables and incorporate any prior knowledge we have about the unknown parameter into a prior probability density for p. Define $p \equiv \boldsymbol{p} \in \mathbb{R}^n$, and let a prior density for $\boldsymbol{p}$ be given by $\pi_{\text{pr}}(\boldsymbol{p})$. Assume that η is distributed according to the density $\pi_\eta(\eta)$. Our goal in Bayesian

[2]This is particularly true for linear inverse problems with Gaussian priors and additive Gaussian noise.

inversion is to construct $\pi_{\boldsymbol{p}|\boldsymbol{d}}(\boldsymbol{p}|\boldsymbol{d})$, the density of our parameters given our particular data, $\boldsymbol{d}$ — we call this the posterior density ($\pi_{\text{post}}(\boldsymbol{p}|\boldsymbol{d})$). This is done using Bayes' rule — we use our data, our knowledge of the forward model and noise to update our prior density. If we have a conditional density of the data given some instance of our parameter, i.e., $\pi_{\boldsymbol{d}|\boldsymbol{p}}(\boldsymbol{d}|\boldsymbol{p})$ (we call this the likelihood density, $\pi_{\text{like}}(\boldsymbol{d}|\boldsymbol{p})$), our posterior density is given by[3]:

$$\pi_{\text{post}}(\boldsymbol{p}|\boldsymbol{d}) \propto \pi_{\text{like}}(\boldsymbol{d}|\boldsymbol{p})\pi_{\text{pr}}(\boldsymbol{p}) \tag{3}$$

There is no universal formula for choosing a prior, and what defines a good prior is often problem specific. The prior should incorporate any knowledge we have, or any nice properties we wish the solution to have in the form of a prior distribution for our parameters. For example, for our advection-diffusion example in section 2, a good prior would ensure sufficient regularity on our parameters for our Bayesian inverse problem to be well-posed. In other words, one choice of good prior would ensure the solution we find is in $\mathcal{L}^2(\Omega)$. The technicalities of how to construct such a prior is beyond the scope of this paper.

Obtaining the likelihood is simple. Assuming we have a particular instance of $\boldsymbol{p} \in \mathbb{R}^n$, all the uncertainty in $\boldsymbol{d}$ is characterized by the uncertainty in η (shifted by $F(\boldsymbol{p})$), implying that:

$$\pi_{\text{like}}(\boldsymbol{d}|\boldsymbol{p}) = \pi_{\eta}(\boldsymbol{d} - F(\boldsymbol{p})) \tag{4}$$

3.2.1 Application to linear inverse problems with Gaussian priors

For the remainder of these notes, we will assume that our parameter-to-observation map is linear and involves a PDE solve. This is the case in our advection-diffusion example since the solution to the PDE depends linearly on the initial condition. Since we focused on finite dimensional inversion, let us further assume that our PDE has been discretized, i.e., we look at $F : V_h \to \mathbb{R}^m$, where V_h is some finite-dimensional subspace of $\mathcal{H}$ (recall that for the advection-diffusion case, you can set $\mathcal{H}$ to be $\mathcal{L}^2(\Omega)$) given by the span of n basis functions (ex. Lagrange basis functions), ϕ_n. Then $\forall p_h \in V_h$, we can write: $p_h = \sum_{i=1}^{n} p_i \phi_i$, and we seek to invert the discretized problem for p_h. Thus, instead of inferring the probability law for our random function p, we focus on characterizing the posterior distribution for our vector of coefficients, $\boldsymbol{p} = [p_1, p_2, \cdots, p_n]$. We now define the fully discretized forward operator to be $\boldsymbol{F} : \mathbb{R}^n \to \mathbb{R}^m$.

We make the assumption that our prior and noise distributions for the discretized variables [4] are Gaussian, i.e., $\boldsymbol{p} \sim \mathcal{N}(\mu_{\text{pr}}, \Gamma_{\text{pr}})$ and $\eta \sim \mathcal{N}(0, \Gamma_{\text{noise}})$ where both Γ_{noise} and Γ_{pr} are symmetric positive definite matrices. Without loss of generality, we take $\mu_{\text{pr}} = 0$. Note that this means $\pi_{\text{pr}} = C \exp\left(-\frac{1}{2}\boldsymbol{p}^T \Gamma_{\text{pr}}^{-1} \boldsymbol{p}\right)$.

Applying (4) to this specific case yields the following likelihood density:

[3] The denominator (density of $\boldsymbol{d}$) is ignored since it is independent of $\boldsymbol{p}$ and acts like a constant.

[4] Generally, for inverse problems governed by PDEs, the approach taken is to formulate the Bayesian inversion in the appropriate Hilbert space $\mathcal{H}$, which requires one to impose a prior probability law for our unknown function, p, of the form $\mu_{\text{pr}} = \mathcal{N}(p_0, \mathcal{C}_0)$ where $p_0 \in \mathcal{H}$ and $\mathcal{C}_0$ is an appropriately chosen covariance operator (often defined to be the inverse of an elliptic differential operator), then discretize. This discretization procedure requires one to work in a mass weighted inner product space for consistency with the infinite-dimensional problem. For details, see [3].

$$\pi_{\text{like}}(\boldsymbol{d}|\boldsymbol{p}) = C \exp\left[-\frac{1}{2}\left(\boldsymbol{d}-\boldsymbol{F}\boldsymbol{p}\right)^* \Gamma_{\text{noise}}^{-1}\left(\boldsymbol{d}-\boldsymbol{F}\boldsymbol{p}\right)\right] \tag{5}$$

Now we can define our posterior probability density:

$$\begin{aligned}
\pi_{\text{post}}(\boldsymbol{p}|\boldsymbol{d}) &\propto \pi_{\text{like}}(\boldsymbol{d}|\boldsymbol{p})\pi_{\text{pr}}(\boldsymbol{p}) \\
&= C \exp\left[-\frac{1}{2}\left(\boldsymbol{d}-\boldsymbol{F}\boldsymbol{p}\right)^* \Gamma_{\text{noise}}^{-1}\left(\boldsymbol{d}-\boldsymbol{F}\boldsymbol{p}\right)\right] \exp\left[-\frac{1}{2}\boldsymbol{p}^T\Gamma_{\text{pr}}^{-1}\boldsymbol{p}\right] \\
&= C \exp\left[-\frac{1}{2}\boldsymbol{p}^T(\boldsymbol{F}^*\Gamma_{\text{noise}}^{-1}\boldsymbol{F}+\Gamma_{\text{pr}})\boldsymbol{p} - \boldsymbol{p}^T\boldsymbol{F}^*\Gamma_{\text{noise}}^{-1}\boldsymbol{d}\right] \\
&= C \exp\left[-\frac{1}{2}(\boldsymbol{p}-\mu_{\text{post}})^T\Gamma_{\text{post}}^{-1}(\boldsymbol{p}-\mu_{\text{post}})\right]
\end{aligned} \tag{6}$$

Where μ_{post} and Γ_{post} are given by:

$$\begin{aligned}
\Gamma_{\text{post}} &= \left(\boldsymbol{F}^*\Gamma_{\text{noise}}^{-1}\boldsymbol{F}+\Gamma_{\text{pr}}^{-1}\right)^{-1} \\
\mu_{\text{post}} &= \Gamma_{\text{post}}\boldsymbol{F}^*\Gamma_{\text{noise}}^{-1}\boldsymbol{d}
\end{aligned} \tag{7}$$

The main take-away from this exercise is that for linear inverse problems with Gaussian priors and Gaussian additive noise, the posterior is Gaussian. Thus, solving the Bayesian inverse problem is equivalent to fully characterizing the posterior covariance and mean. Unfortunately, for many problems governed by PDEs this is not always as easy as it sounds. The posterior covariance matrix requires the inversion of an operator involving a forward and adjoint PDE solve. It is often too expensive to have an explicit matrix form of our forward (or adjoint) operator — but, given any parameter, we can compute the corresponding observations. Thus we require a matrix-free algorithm for the computation of the inverse. Our operator is symmetric positive definite (SPD), so we can certainly use conjugate gradient to obtain the action of the inverse on any vector in $\mathbb{R}^n$, but this requires many forward and adjoint PDE solves. Moreover, if one wished to fully characterize the posterior, one would need to compute the action of the inverse on the identity matrix in $\mathbb{R}^n$. Since n depends on the discretization of the PDE, this could quickly become very expensive.

4 Optimal experimental design

The general question underlying optimal experimental design for inverse problems is how to design an experiment to optimally infer an unknown parameter. How to define the "design" as well as the concept of optimality is problem-specific. Here we focus on Bayesian linear problems governed by PDEs. In particular, we restrict ourselves to finding a subset of a finite number of sensors scattered over our domain. This section serves to give an overview of the method described in [1] and thus we follow the structure closely.

4.1 Problem formulation for linear Bayesian inverse problems

The problem is as follows: given a time-dependent PDE and a domain Ω with $N_s \in \mathbb{Z}$ candidate sensors at locations x_i for $i = 1, \cdots, N_s$ for data collection at N_τ finite times (we assume that the sensor locations are fixed for all time), choose as few target locations as possible for optimal reconstruction of some unknown input to the PDE. The input could be an initial condition, a boundary value, a diffusion coefficient, or any combination of the former. Referring back to 2, the input is a discretized representation of an initial condition. This means that $\boldsymbol{F} \in \mathbb{R}^{N_s N_\tau \times n}$.

To incorporate the "design" for our problem, we choose to incorporate a weight, w_i to each sensor at location x_i. Ideally, we want $w_i = 1$ if we choose to keep the sensor at location x_i, and $w_i = 0$ otherwise. However, combinatorial optimization problems are NP-hard, so we are forced to relax the problem by choosing weights w_i in the interval $[0, 1]$. Each different set of weights $\boldsymbol{w} = [w_1, w_2, \cdots, w_{N_s}]$ corresponds to a different design. There is no correct physical way to incorporate a fractional weight at a particular sensor, but the relaxation is done to make the problem computationally feasible. An ideal algorithm would approximate the optimal binary weight design as best as possible.

Keeping this weight vector $\boldsymbol{w}$ in mind, how do we incorporate this design into our Bayesian inverse problem? For this, define a block diagonal matrix with N_τ blocks, $\boldsymbol{W} \in \mathbb{R}^{N_s N_\tau \times N_s N_\tau}$ with the weights $\boldsymbol{w} = [w_1, w_2, \cdots, w_{N_s}]$ on the diagonal of every $N_s \times N_s$ sub-block. We now re-define our forward map $\boldsymbol{F}$ to include this weight matrix, $\widehat{\boldsymbol{F}} = \boldsymbol{W}^{\frac{1}{2}} \boldsymbol{F}$. In other words, our forward map takes in a parameter, $\boldsymbol{p} \in \mathbb{R}^n$, maps it to our observations at *all* N_s sensors at N_τ times, and the weight matrix assigns a weight $\sqrt{w_i}$ to each of the observations at sensor location x_i. With this notation in mind, we reconstruct the posterior covariance matrix in (7) to incorporate this new weight forward map (we take $\Gamma_{\text{noise}} = I$ for algebraic simplicity).

$$\begin{aligned} \Gamma_{\text{post}}(\boldsymbol{w}) &= \left(\widehat{\boldsymbol{F}}^* \widehat{\boldsymbol{F}} + \Gamma_{\text{pr}}^{-1} \right)^{-1} \\ &= \left(\boldsymbol{F}^* \boldsymbol{W} \boldsymbol{F} + \Gamma_{\text{pr}}^{-1} \right)^{-1} \\ &= \Gamma_{\text{pr}}^{\frac{1}{2}} \left(\Gamma_{\text{pr}}^{\frac{1}{2}} \boldsymbol{F}^* \boldsymbol{W} \boldsymbol{F} \Gamma_{\text{pr}}^{\frac{1}{2}} + \boldsymbol{I} \right)^{-1} \Gamma_{\text{pr}}^{\frac{1}{2}} \end{aligned} \tag{8}$$

A natural choice for optimal design for this particular problem is one which minimizes the average posterior variance, i.e., the trace of the posterior covariance matrix ([1] refers to this as A-optimality). Minimizing the posterior variance allows us to choose a design which maximizes how much we have learned and our certainty in the solution.

Intuitively, the more locations available for data collection, the more we can learn, i.e., we have less variance in the posterior distribution[5]. However, keeping in mind the real-world monetary and physical constraints on sensor placement, we seek out sparse solutions, i.e., ones for which many of the weights are set to zero.

Keeping the above in mind, the minimization problem we solve to obtain the optimal design is:

$$\boldsymbol{w}^* = \text{argmin}_{\boldsymbol{w} \in \mathbb{R}^{N_s}} \, \text{trace} \left[\Gamma_{\text{post}}(\boldsymbol{w}) \right] + \gamma \psi \left(\boldsymbol{w} \right) \quad \text{subject to } w_i \in [0, 1] \, \forall i \tag{9}$$

[5]This statement can also be seen mathematically by looking at the last equality in (10). If a sensor is not included, and the weight is zero, then $\boldsymbol{D}_{ii} = 0$, so we do not learn in the direction $\boldsymbol{U}_i$.

Here $\psi(\boldsymbol{w})$ is a sparsity inducing penalty function, and γ controls the degree of sparsity. One of the standard ways to enforce sparsity is to introduce a penalty induced by the ℓ_1 norm, $\psi(\boldsymbol{w}) = \mathbf{1}^T\boldsymbol{w}$. This choice of norm, as opposed to $\ell_i = \sum_{j=1}^{n} w_j^i$ for $i > 1$, ensures that solutions further away from zero are penalized more heavily 2. A downside to this approach is that it will not yield $w_i \in \{0, 1\}$, and it is not immediately clear how to interpret a solution with continuous weights. We can choose to include all sensors with non-zero weights, but this will most likely be less sparse than optimal.

To avoid this ambiguity, one would like to set the penalty function to be the number of non-zero sensor weights (or the ℓ_0 "norm" of the weight vector). This method directly leads to binary weight vectors. The non-rigorous/intuitive explanation for this is that the penalty cost of having a fractional non-zero weight is the same as having a weight of one, but the trace of the variance can only get smaller as the weight is increased to one (the larger the weight of a sensor, the more we can "learn" from the observations taken from that sensor). Unfortunately, this penalty is non-convex and not differentiable, thus some way to approximate it is needed. One approximation approach is discussed in detail in [1]. The idea here is to iteratively approximate the ℓ_0 solution by solving (9) with a sequence of smooth penalty functions which converge to the ℓ_0 "norm".

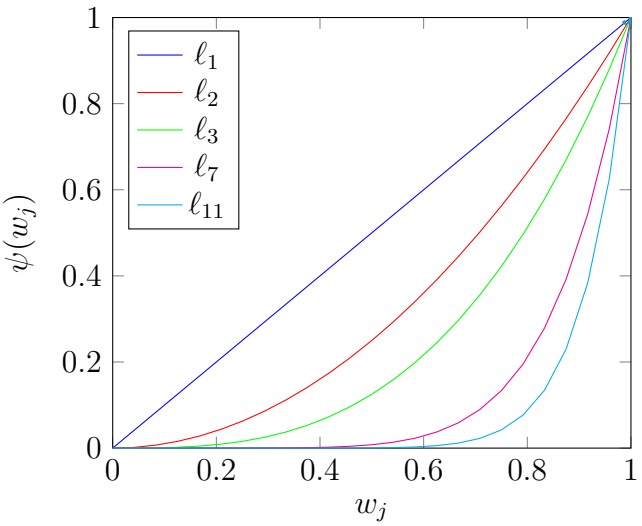


Figure 2: Cost of setting an individual sensor weight between 0 and 1 for various penalty functions. We can see that the ℓ_1 penalty incurs the highest cost and thus favors sparse solutions.

4.2 Computational Challenges

The computational difficulties of solving the minimization problem (9) all stem from the fact that $\boldsymbol{F}$ is a discretized PDE operator.

Standard minimization techniques, such as Newton's method or BFGS, all require multiple applications of the gradient (and potentially Hessian) of (9). This means that for each iteration, we have to solve a new Bayesian inverse problem to compute $\Gamma_{\text{post}}(\boldsymbol{w})$. For PDE models which are expensive to compute, or ones for

which the discretized parameters live in very high dimensional spaces, this can become infeasible. For standard inverse problems, we can compute an approximation to Γ_{post} in terms of a low-rank negative definite update to the prior covariance. However, because the Γ_{post} for the OED problem depends on the vector we are trying to minimize over, this does not solve our problem here. Instead, we can find a low-rank approximation to the prior-preconditioned forward map, i.e., $\widetilde{\boldsymbol{F}} = \boldsymbol{F}\Gamma_{\text{pr}}^{\frac{1}{2}}$ in terms of a low-rank singular value decomposition. Having a decomposition of $\widetilde{\boldsymbol{F}}$ rather than $\boldsymbol{F}$ will become convenient later, but it has an additional benefit of being more efficient to compute, since the prior is often smoothing, thus $\widetilde{\boldsymbol{F}}$ has a quicker decaying eigenspectrum. This step can be done offline, then in the minimization step, a forward or adjoint PDE solve is reduced to simple matrix vector computations.

Another issue is how to compute the trace and its gradient. The prior covariance operator often involves the solution of a PDE, so we do not have Γ_{pr}^{-1} in explicit form. Thus we can't directly compute Γ_{post}, let alone apply the trace operator to it. A way to fix this is to use randomized trace estimation. For this, the idea is to use N_{tr} Gaussian random vectors to approximate the trace using Monte-Carlo estimates. Given N_{tr} Gaussian random vectors, z_i, the trace of Γ_{post} is approximated by $\frac{1}{N_{\text{tr}}}\sum_{i=1}^{N_{\text{tr}}} {z_i}^T\Gamma_{\text{post}}z_i$[6]. For this method, one only needs to be able to apply an operator to a vector. This form can also be used to approximate the gradient of the functional in a form that only requires applications of Γ_{post} to a vector.

We are still left with the issue of how to apply Γ_{post}. Rather than solving $\Gamma_{\text{post}}^{-1}y_i = z_i$ for each individual random vector using some iterative scheme, one can find a low-rank approximation to $\widetilde{\boldsymbol{F}}^*\boldsymbol{W}\widetilde{\boldsymbol{F}} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{U}^T$. Sherman-Morrison formula can then be used to rewrite Γ_{post} in a manageable form. More rigorously, we do the following computation:

$$\begin{aligned}\Gamma_{\text{post}} &= \Gamma_{\text{pr}}^{\frac{1}{2}}\left(\Gamma_{\text{pr}}^{\frac{1}{2}}\boldsymbol{F}^*\boldsymbol{W}\boldsymbol{F}\Gamma_{\text{pr}}^{\frac{1}{2}} + \boldsymbol{I}\right)^{-1}\Gamma_{\text{pr}}^{\frac{1}{2}} \\ &= \Gamma_{\text{pr}}^{\frac{1}{2}}\left[\boldsymbol{I} - \boldsymbol{U}\boldsymbol{\Sigma}^{\frac{1}{2}}\left(\boldsymbol{I}+\boldsymbol{\Sigma}\right)^{-1}\boldsymbol{\Sigma}^{\frac{1}{2}}\boldsymbol{U}^T\right]\Gamma_{\text{pr}}^{\frac{1}{2}} \\ &= \Gamma_{\text{pr}} - \boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^T\end{aligned} \tag{10}$$

Here, $\boldsymbol{D}$ is defined to be a diagonal matrix with entries $\boldsymbol{D}_{ii} = \frac{\boldsymbol{\Sigma}_{ii}}{\boldsymbol{\Sigma}_{ii}+1}$ Applying Γ_{post} now just requires some matrix vector multiplication.

5 Results for advection-diffusion example

Here we look back to our model equation (1) presented in section 2, and present some numerical results. In particular, we present a specific choice of prior and plot samples from both the prior and posterior distribution to better illustrate the Bayesian approach. Additionally, we solve the OED problem with regularized ℓ_0 sparsification, a method of approximating the ℓ_0 penalty introduced in [1], and show the optimal sensor locations. Applying the forward operator, $\boldsymbol{F}$ involves calling a finite element method code to solve the advection-diffusion equation and interpolate the discretized solution to observations at N_s sensor locations at times $t_1, \cdots, t_{N_\tau}$.

For the numerical results, κ was set to 0.001, and T, the final time was set to 4. As described in [1], the velocity field $\boldsymbol{v}$ was obtained through a solution of a steady-

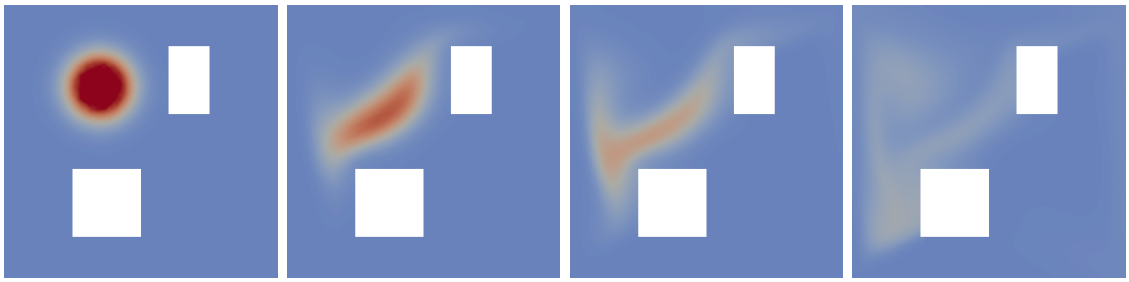
[6]For details on the theory behind this, see [2]

state Navier-Stokes equation. The induce a Gaussian prior on u_0, $u_0 \sim \mathcal{N}(\mu_0, \mathcal{C}_0)$, with $\mathcal{C}_0 = \mathcal{A}^{-2}$. Here, $\mathcal{A}$ is an elliptic differential operator. The PDE that $\mathcal{A}$ represents is defined as follow: $\forall y \in \mathcal{L}^2(\Omega)$, the solution $v = \mathcal{A}^{-1}y$ satisfies:

$$\int_\Omega \alpha \nabla v \cdot \nabla p + \beta v p dx = \int_\Omega y p dx \quad \forall p \in H^1\Omega$$

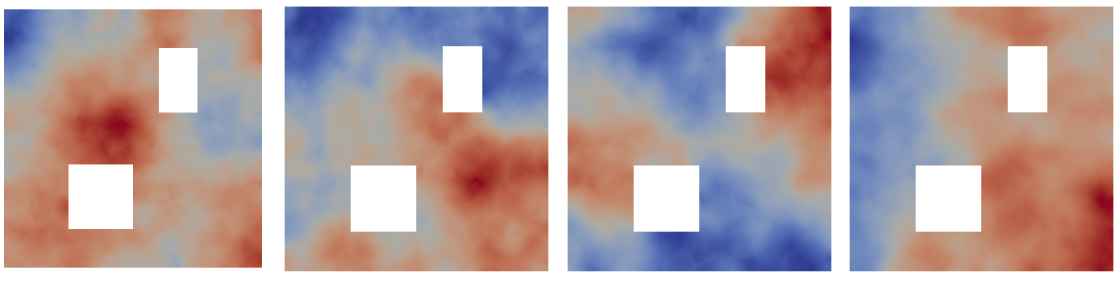
Fist we solve the discretized inverse problem to obtain a posterior density. Figure 3 shows the true initial condition and the time evolution. Figure 4 shows some samples obtained from the bilaplacian prior distribution.

The optimal fifteen locations for sensor placement are depicted in figure 5. To compare how well these fifteen sensors do, we show a few various numerical results. First, one can look at the difference in the samples obtained from the posterior using the optimal sensor locations (5) versus all 124 possibilities (5). While we can clearly see that the samples obtained using just 15 sensors are not very different from the ones obtained using all candidate sensor locations, to further analyze how well these optimal 15 sensors do, the posterior pointwise variance is compared to the posterior pointwise variance obtained from using all 124 sensors 8. Finally we plot the posterior pointwise variance obtained using the optimal 15 sensors versus two possibilites of fifteen randomly chosen locations for sensor placement (9). It is interesting to note that for this particular problem, the difference in the posterior pointwise variance obtained using the optimal sensors versus randomly chosen locations decreases as the number of possible sensors you can place increases.



$t = 0$ $t = 1$ $t = 2$ $t = 4$

Figure 3: Evolution of the advective-diffusive transport. Figure taken from [5].



$t = 0$ $t = 1$ $t = 2$ $t = 4$

Figure 4: Samples taken from the bilaplacian prior density for the initial condition in the advection-diffusion problem. Figures obtained using [5].

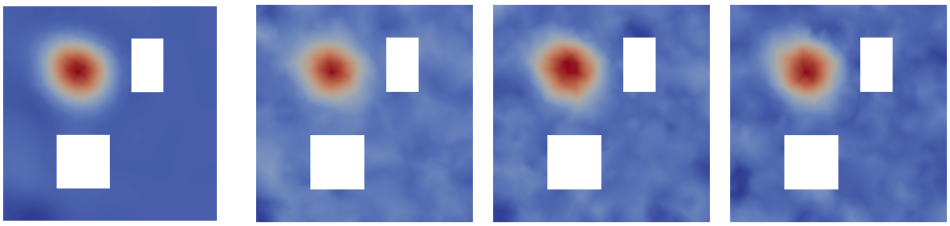


Figure 5: Samples taken from the posterior density for the initial condition in the advection-diffusion sample problem. Here all the possible sensors are included, i.e., weight vector $\boldsymbol{w} = \mathbf{1}^T$ (first three images). The true initial condition is in the leftmost image. Comparing these samples to the ones taken from the prior in figure 4, we can see that we have done a pretty good job of reconstructing the initial condition. [5]

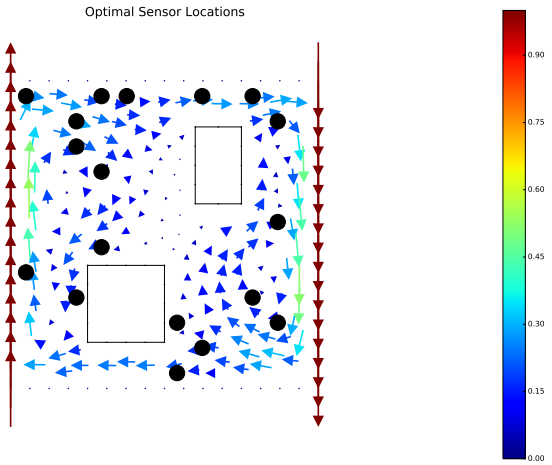


Figure 6: Optimal locations of 15 sensors obtained using regularized ℓ_0 sparsifications. The sensor locations are depicted as black dots and they are plotted over the velocity field used in the advetion-difussion equation.

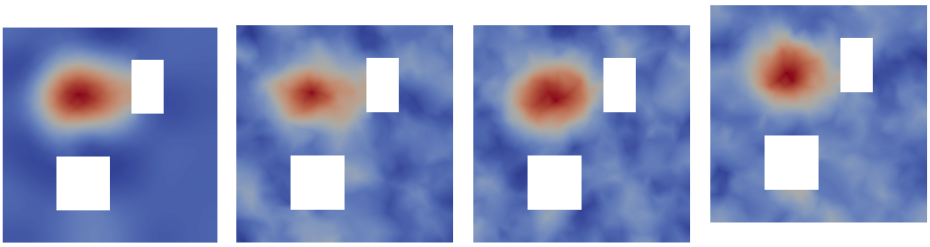


Figure 7: Mean (left-most) and samples (last 3 columns) from the posterior using the optimal 15 sensors obtained, depicted in figure 5.

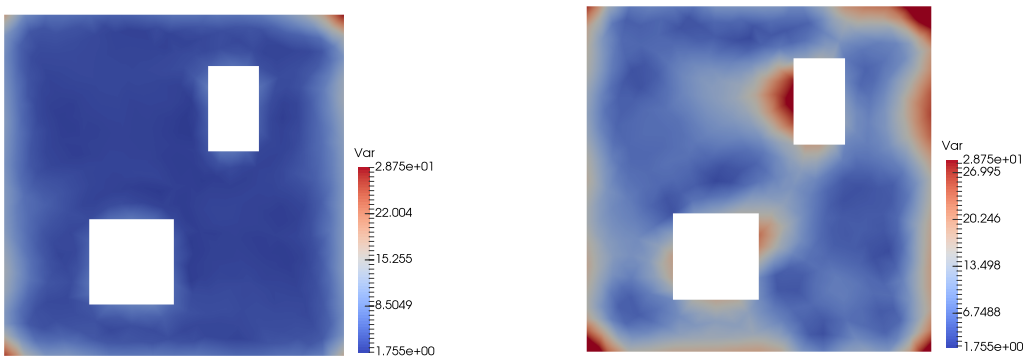


Figure 8: Posterior pointwise variance using all 124 sensors (left), and one obtained using 15 sensors (right)

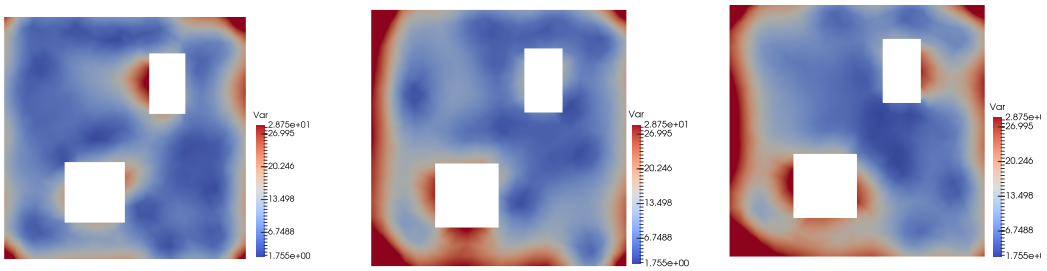


Figure 9: Posterior pointwise variance for optimal 15 (left), and two randomly chosen locations for 15 sensors (middle and right)

6 Conclusions

This paper has focused on inverse problems governed by linear forward operators. We have presented the Bayesian approach — one method used for solving these problems — and touched upon some of the difficulties. Even for this limited subset of inverse problems, characterizing a posterior distribution can get intractable when the problem is governed by a forward operator involving a PDE solve. We have also presented one optimal experimental design problem, focusing on designs which minimize the average posterior variance. Of course, there are other variations of "optimality" one can consider which we have not discussed. Two examples of alternative optimal designs include: ones which minimze the determinant of the posterior covariance, and ones that minimize the largest eigenvalue of the posterior covariance matrix.

The area of non-linear inquiry presents its own set of methods and challenges and is beyond our limited scope. Even in our discussed subset of linear forward operators though, there is further work to be done. One can imagine, for example, that many linear inverse problems motivated by real-life phenomena have some inherent uncertainty. To illustrate, in our NYC advective-diffusive example presented in section 2, we do not know what the exact wind field will be in the event of a chemical attack. It is thus useful to construct OED methods which give optimal results under uncertain inputs. This problem, while very similar, yields an added layer of difficulties. If we have a density for the input causing the uncertainty, we can

sample it to obtain some proposed velocities and use them to construct an averaged version of our forward map. For this problem, care must be taken to find an efficient decomposition of this new averaged operator.

It is hoped that this paper has touched on a sufficient overview of the background required to tackle problems in the field of OED and highlighted some of the interesting features and challenges in this area of research. Indeed, the area of inverse problems as a whole necessitates an understanding and comfort with the interplay of many branches of mathematics and presents a significant number interesting avenues for researchers to explore further.

References

[1] Alen Alexanderian, Noemi Petra, Georg Stadler, and Omar Ghattas. A-optimal design of experiments for infinite-dimensional bayesian linear inverse problems with regularized \ell_0-sparsification. *SIAM Journal on Scientific Computing*, 36(5):A2122–A2148, 2014.

[2] Haim Avron and Sivan Toledo. Randomized algorithms for estimating the trace of an implicit symmetric positive semi-definite matrix. *Journal of the ACM (JACM)*, 58(2):8, 2011.

[3] Tan Bui-Thanh, Omar Ghattas, James Martin, and Georg 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(6):A2494–A2523, 2013.

[4] Masoumeh Dashti and Andrew M Stuart. The bayesian approach to inverse problems. *Handbook of Uncertainty Quantification*, pages 311–428, 2017.

[5] U. Villa, N. Petra, and O. Ghattas. hIPPYlib: an Extensible Software Framework for Large-scale Deterministic and Linearized Bayesian Inversion. 2016.