Statistical Finite Elements via Interacting Particle Langevin Dynamics

Alex Glyn-Davies, Connor Duffin, Ieva Kazlauskaite, Mark Girolami, Ö. Deniz Akyildiz

May 28, 2025

Abstract

In this paper, we develop a class of interacting particle Langevin algorithms to solve inverse problems for partial differential equations (PDEs). In particular, we leverage the statistical finite elements (statFEM) formulation to obtain a finite-dimensional latent variable statistical model where the parameter is that of the (discretised) forward map and the latent variable is the statFEM solution of the PDE which is assumed to be partially observed. We then adapt a recently proposed expectation-maximisation like scheme, interacting particle Langevin algorithm (IPLA), for this problem and obtain a joint estimation procedure for the parameters and the latent variables. We consider three main examples: (i) estimating the forcing for linear Poisson PDE, (ii) estimating the forcing for nonlinear Poisson PDE, and (iii) estimating diffusivity for linear Poisson PDE. We provide computational complexity estimates for forcing estimation in the linear case. We also provide comprehensive numerical experiments and preconditioning strategies that significantly improve the performance, showing that the proposed class of methods can be the choice for parameter inference in PDE models.

1 Introduction

Statistical estimation methods for partial differential equation (PDE) models are of significant recent interest as such approaches offer powerful tools to analyse complex systems which can be jointly described through both mechanistic and data-driven components. They allow for the estimation of partially or indirectly observed quantities-of-interest, which, depending on the setting may be a model input parameter (*inversion*) [45, 46], or, a model state (*data assimilation*) [42]. In this work we are interested in solving these problems jointly, i.e., the joint parameter and state estimation problem within the context of static stochastic PDEs. We aim at tackling the problem of inference for PDEs using a recent statistical construction termed the statistical finite element method (statFEM) [4, 18, 28], which provides a statistical model of the PDE solution. Using this statistical model, we aim at solving the joint state and parameter estimation problem using the interacting particle Langevin algorithm (IPLA) [3], a recently proposed class of statistical estimation methods for latent variable models.

To set up the context, consider the following stochastic PDE with the differential operator $\mathcal{L}_\theta$ parameterised by θ, and forcing f

$$\mathcal{L}_\theta u = f + \epsilon, \quad \epsilon \sim \mathcal{GP}(0, k) \tag{1}$$

where k is a (regular enough) kernel function (see section 2.1 for details). A data generating process can then be defined through the continuous *observation operator* $\mathcal{H}\colon L^2(\Omega) \to \mathbb{R}^{n_y}$

$$\mathbf{y} = \mathcal{H}(u) + \mathbf{r}, \quad \mathbf{r} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}). \tag{2}$$

The equations (1)–(2) specify a statistical model, which can be used to infer the latent PDE solution u. This is a standard problem, having been the subject of much recent attention with, for example, the

statFEM [4, 18, 28]. While these methods can be generically used for sampling the parameters jointly, there is a gap in the literature for dedicated methods to obtain point estimates of the parameters directly, which may avoid a separate procedure to get maximisers from samples. Denote by $\mathbf{z}$ these unknown parameters, which may comprise any unknown model components, e.g. PDE parameters θ, or the forcing function f. Discretising (1) using statFEM, we obtain a model state vector $\mathbf{u}$, and conditional distribution $p(\mathbf{u}|\mathbf{z})$. To estimate the parameters of the PDE (that are denoted as $\mathbf{z}$), we aim at maximising the marginal likelihood by solving the following optimisation problem

$$\mathbf{z}_{\text{MMLE}} = \underset{\mathbf{z}\in\mathcal{Z}}{\operatorname{argmax}} \log p(\mathbf{y}|\mathbf{z}), \quad \text{where} \quad p(\mathbf{y}|\mathbf{z}) := \int p(\mathbf{y}|\mathbf{u})\, p(\mathbf{u}|\mathbf{z}) \mathrm{d}\mathbf{u}, \tag{3}$$

which is termed the marginal maximum likelihood estimation (MMLE) problem. When we include a prior distribution in the model, the marginal *maximum a posteriori* (MMAP) problem can be solved by maximising the marginal posterior distribution

$$\mathbf{z}_{\text{MMAP}} = \underset{\mathbf{z}\in\mathcal{Z}}{\operatorname{argmax}} \log p(\mathbf{z}, \mathbf{y}), \quad \text{where} \quad p(\mathbf{z}, \mathbf{y}) := \int p(\mathbf{y}|\mathbf{u})\, p(\mathbf{u}|\mathbf{z}) p(\mathbf{z}) \mathrm{d}\mathbf{u}. \tag{4}$$

In this paper we will focus on this estimation, using the so-called IPLA, which also provides us with estimates of the posterior $p(\mathbf{u}|\mathbf{y}, \mathbf{z})$. Our choice of IPLA is motivated by its theoretical guarantees, which allow us to conduct some analysis in our proposed model setting. To set the scene for this work, we now go through a review of the relevant literature.

1.1 Statistical Finite Elements

Introduced in [28], the statFEM was developed for calibrating finite element models using observational data. Uncertainties from model misspecification and observational noise are assumed, and are used to construct a statistical model describing how the stochastic finite element method (FEM) solution generates the data. A Bayesian perspective is taken, and the prior over the FEM solution is updated based on observations via posterior inference. This has been extended to time-varying PDEs in [18], where filtering techniques are used to sequentially update the posterior distribution. The Bayesian framework also allows for estimation of random field parameters based on the marginal likelihood of the data [18, 26, 28]. Recently, theoretical advances have been made, including error analysis in [33] and convergence analysis with mesh refinement in [39]. Langevin dynamics has been proposed as a method for efficient sampling from the posterior distribution [4], which includes analysis of convergence of the sampler to the target distribution.

1.2 Marginal MLE and MAP

In this section, we first review the statistical literature pertaining to the MMLE. The estimation problem in (3) can be solved with the classical expectation-maximisation (EM) algorithm [16], or, the stochastic/Monte Carlo variants thereof [11, 49]. For a significant number of models, the EM can be implemented via stochastic approximations when the posterior distribution is tractable and easy to sample from. When this sampling is infeasible, the E-step can be approximated using Markov chain Monte Carlo (MCMC) methods [15, 5], which provide a rich framework for bespoke inference. Recent extensions have considered *unadjusted* MCMC, whereby the Metropolis adjustment is skipped for computational simplicity. Perhaps the most notable of these is the stochastic optimization via unadjusted Langevin (SOUL) algorithm [14], which makes use of the unadjusted Langevin algorithm (ULA) [12, 19], to solve MMLE problems. Through Fisher's identity the SOUL approach leverages a stochastic approximation, iteratively running the E-step with a ULA chain, then using these samples to compute the M-step. The main bottleneck of this method is that it requires a unique Markov chain to be run to compute the E-step, for each M-step.

A similar approach is taken in [34], which replaces the coordinate-wise procedure of [14] with an interacting particle system. This method constructs N particles for the latent variables (instead of running a chain in time) and builds a procedure in the joint space of N latent variables and parameters of interest. The authors

demonstrate favourable convergence properties and computational advantages over SOUL, with additional theoretical results [10]. In this work, we follow a similar approach, closer to the standard Langevin dynamics, namely the IPLA [3]. This modifies the particle gradient descent of [34] through modifying the parameter update by noising it, hence forming a stochastic differential equation (SDE) instead of a system of mixed SDEs and ordinary differential equation (ODE) as in [34]. Taking such an approach enables straightforward nonasymptotic results for estimating $\mathbf{z}$, makes the analysis akin to the standard analysis of Langevin diffusions. This similarity allows us to adapt results from standard discretisation error analysis of the Langevin diffusion, which we leverage in this work.

The problem of joint state and parameter estimation has received attention from various communities, including physics [1], chemistry [17], electrical engineering [47, 48], data assimilation [36, 25], and statistics [32]. For the extension to PDE systems, Bayesian inverse problems for PDEs tackle the problem of parameter estimation (i.e. $\mathbf{z} = \theta$) from data or *inversion*, but typically assume a *known* (and deterministic) PDE forward model, which is restrictive in the modelling of uncertainties. Estimation of PDE forcing/source terms from partial observations (i.e. $\mathbf{z} = f$) is relevant to climate modelling, for example the estimation of greenhouse gas sources from satellite observations [37] or identification of pollutant sources [6]. This is also of interest to the engineering community, for example in structural health monitoring for estimating unknown forcing from sensor measurements [22]. By constructing a probabilistic forward model based on our assumed PDE, we allow for principled accounting for model misspecification, and joint estimation of parameters and uncertainty quantification for our PDE solution.

1.3 Our contribution

Previous work has shown that sampling from (1), and its resultant posterior distribution can be efficiently done using ULA [4]. However, given observations and unknown parameters, the resulting PDE model is a latent variable statistical model where the inference of parameters and the latent field requires more than just sampling algorithms. We build on the interacting particle solutions which are shown to be efficient for MMLE problems [3, 34, 21, 38].

Through the use of IPLA in combination with statFEM, this paper provides a novel algorithmic framework to conduct PDE state and parameter estimation, studied in the context of elliptic PDEs. Convergence of the schemes is analysed, and we provide thorough numerical evidence of their performance. More specifically:

- We develop an algorithm for forcing estimation (section 3.1) and solving the inverse problem (section 3.2) in section 3, which is applied to the linear Poisson PDE. Our choice of Poisson PDEs is for demonstration since our algorithms can be similarly adapted to other PDEs.

- We develop preconditioning strategies within this section to improve the condition number of the problem. We also introduce warm-start strategies to avoid the numerical difficulties caused by the ill-conditioning of the problem.

- We then provide, in section 4.1, the rate of convergence for the forcing estimation procedure presented in section 3.1. In particular, in Theorem 1, we prove a nonasymptotic bound for the IPLA procedure as applied to the forcing estimation problem and provide complexity estimates in Remark 1. Our analysis shows that, provided that the number of particles N is chosen such that $N = \mathcal{O}(\varepsilon^{-2} n_u)$ and the step-size γ of our method is chosen so that $\gamma = \mathcal{O}(\varepsilon^2 n_u^{-1} \kappa^{-2})$ where n_u is the degrees of freedom and κ is the condition number of the problem, our method provably achieves ε error in $\widetilde{\mathcal{O}}(n_u \kappa^2 \varepsilon^{-2})$ steps. This also shows why preconditioning is crucial for our case.

- We provide experiments to demonstrate the order of convergence proved in Theorem 1 using an analytical example, as well as the benefit preconditioning through computation of the condition number (section 5.1). We show the benefit of using a warm-start for solving the inverse problem in section 5.2.

- We develop an algorithm for forcing estimation in the nonlinear statFEM setting (section 6.1), which we apply to the nonlinear Poisson PDE (section 6.2). We compare three different nonlinear approximations for forcing estimation in section 6.3.

This paper is organised as follows: in section 2 we introduce the statistical finite element method for linear PDEs (section 2.1) before introducing Langevin dynamics for statFEM (section 2.2) and the interacting particle Langevin algorithm (section 2.3). Section 3 shows how the statFEM construction can be used to embed the probabilistic model for the linear Poisson PDE within the IPLA for MMLE/MMAP estimation. The algorithms and preconditioners are given in section 3.1 for forcing estimation, and section 3.2 for solving the inverse problem. Theoretical results are derived in section 4. Experimental results for the linear Poisson PDE are given in section 5, including the forcing estimation (section 5.1), and the inverse problem (section 5.2) results. In section 6, we extend the methodology to nonlinear statFEM. Section 6.1 derives the form of the statFEM prior for the nonlinear Poisson PDE, and section 6.2 outlines three different approximations to this prior for use in the IPLA algorithm. Forcing estimation results are given in section 6.3. We conclude the paper in section 7.

2 Technical Background

2.1 Statistical Finite Elements for Linear PDEs

The statistical finite element method constructs a prior distribution over the finite element coefficients of a PDE solution, based on an additive Gaussian process (GP) forcing error. For linear differential operators, this induced distribution is itself a Gaussian, and can be found in closed form. Consider an elliptic PDE with homogeneous Dirichlet boundary conditions [24], with linear differential operator $\mathcal{L}_\theta$, parametrised by θ, acting on the solution $u := u(x)$ defined over some domain $x \in \Omega$,

$$\begin{aligned} \mathcal{L}_\theta u &= f + \epsilon, \quad x \in \Omega \\ u &= 0, \quad x \in \partial\Omega, \end{aligned} \tag{5}$$

with the forcing $f \in L^2(\Omega)$, and the additive GP noise by $\epsilon \sim \mathcal{GP}(0, k)$ with covariance function $k\colon \Omega \times \Omega \to \mathbb{R}$. To convert the strong form of the PDE to the weak form we multiply both sides by a test function $v \in V$, where V is an appropriate function space (e.g. $H_0^1(\Omega)$, the Sobolev space of square-integrable functions with square integrable first-order weak derivatives) and integrate over the domain,

$$\int_\Omega (\mathcal{L}_\theta u) v \mathrm{d}x = \int_\Omega (f + \epsilon) v \mathrm{d}x, \quad \forall v \in V. \tag{6}$$

For the elliptic PDE, the order of differentiation can be reduced via integration by parts yielding a bilinear form $\mathcal{A}_\theta(u, v)$. This gives the following variational form

$$\mathcal{A}_\theta(u, v) = \langle f + \epsilon, v \rangle, \quad \forall v \in V, \tag{7}$$

where $\langle \cdot, \cdot \rangle$ is the $L^2(\Omega)$ inner product. The FEM method forms a discrete approximation with a finite-dimensional set of basis functions defined over the domain, $\{\phi_j(x)\}_{j=1}^{n_u}$. We search for solutions in the subspace $V_h \subset V$ defined by the span of these basis functions $V_h = \operatorname{span}\{\phi_j(x)\}_{j=1}^{n_u}$, where the parameter h refers to the degree of mesh-refinement of the domain. The basis function expansion $u_h(x) = \sum_{j=1}^{n_u} \hat{u}_j \phi_j(x)$ provides a finite-dimension variational problem

$$\mathcal{A}_\theta(u_h, \phi_j) = \sum_{i=1}^{n_u} \hat{u}_i \mathcal{A}_\theta(\phi_i, \phi_j) = \langle f, \phi_j \rangle + \langle \epsilon, \phi_j \rangle, \quad \forall j \in \{1, \ldots, n_u\}, \tag{8}$$

which can be rewritten as the following finite-dimensional linear system

$$\mathbf{A}_\theta \mathbf{u} = \mathbf{b} + \mathbf{e}, \quad \mathbf{e} \sim \mathcal{N}(\mathbf{0}, \mathbf{G}), \tag{9}$$

with $\mathbf{u} = [\hat{u}_1, \hat{u}_2, \ldots, \hat{u}_{n_u}]^\top$, $[\mathbf{A}_\theta]_{ij} = \mathcal{A}_\theta(\phi_i, \phi_j)$, $[\mathbf{b}]_i = \langle f, \phi_i \rangle$, $[\mathbf{G}]_{ij} = \langle \phi_i, \langle k(\cdot,\cdot), \phi_j \rangle \rangle$. Since Equation (9) is linear and the additive noise is Gaussian, we have a closed form expression for the prior over FEM coefficients that is also Gaussian and defined by

$$p(\mathbf{u}|\theta, \mathbf{b}) = \mathcal{N}\left(\mathbf{A}_\theta^{-1}\mathbf{b}, \mathbf{A}_\theta^{-1}\mathbf{G}\mathbf{A}_\theta^{-\top}\right). \tag{10}$$

Provided the bilinear form $\mathcal{A}_\theta$ is coercive on V the problem is well-posed, the stiffness matrix $\mathbf{A}_\theta$ will be positive definite, implying its inverse exists (see e.g. [23][Remark 2.20]). For setting the homogeneous Dirichlet boundary conditions, we set the rows corresponding to boundary nodes to the row of the identity matrix, and the forcing vector to zero. This can be adapted for inhomogeneous boundary conditions via boundary condition lifting (see e.g. [30][Page 8]), which adapts the entries of the right-hand side vector to the non-zero boundary condition. We choose to constrain the covariance function k to be zero on the boundary, so that boundary conditions are imposed exactly. To achieve this, we form an approximation to the GP, as seen in [44], which has the benefit in our case of being constrained to zero covariance on the boundary. The eigenfunctions of the Laplacian operator (essentially the vibrational modes of the domain) form the basis, which are weighted based on the spectrum of the covariance function. This approximation is given by

$$k(x, x') \approx \sum_{l=1}^{m} S\left(\sqrt{\lambda_l}\right) g_l(x) g_l(x') \tag{11}$$

where the function $g_l(x) = \sum_{l=1}^{n_u} \tilde{g}_{li}\phi_i(x)$ is the L^2-normalised FEM approximation of the l^{th} eigenfunction of the Laplacian operator, λ_l the corresponding eigenvalue, and m the rank of the approximation. The weighting function $S(\cdot)$ is the spectral density of the covariance function k. Note this requires an isotropic covariance function, i.e. such that we can write $k(x, x') = k(\|x - x'\|)$. This is detailed in Appendix F.

2.2 Statistical Finite Elements via Langevin Dynamics

Having derived the prior density, we can construct a Langevin SDE that takes samples from this distribution by defining the potential $\Phi_\theta(\mathbf{u}, \mathbf{b}) := -\log p(\mathbf{u}|\theta, \mathbf{b})$. The Langevin SDE with this potential as the drift has the prior distribution as its limiting distribution:

$$\mathrm{d}\mathbf{u}_t = -\nabla_{\mathbf{u}}\Phi_\theta(\mathbf{u}_t, \mathbf{b})\mathrm{d}t + \sqrt{2}\,\mathrm{d}\mathbf{B}_t, \tag{12}$$

where $(\mathbf{B}_t)_{t\geq 0}$ is a n_u-dimensional Brownian motion. The Euler-Maruyama discretisation is used to simulate from this Langevin SDE, which approximates the transition density between successive time-points with a Gaussian distribution [40]. The associated discrete-time Markov chain

$$\mathbf{u}_{k+1} = \mathbf{u}_k - \gamma\nabla_{\mathbf{u}}\Phi_\theta(\mathbf{u}_k, \mathbf{b}) + \sqrt{2\gamma}\,\boldsymbol{\zeta}_{k+1} \tag{13}$$

where $\{\boldsymbol{\zeta}_k\}_{k\in\mathbb{N}}$ is a sequence of i.i.d. standard Gaussian random variables, and $\gamma > 0$ controls the degree of time-discretisation. This discretisation induces a bias, the degree of which is determined by the step-size γ. For some applications this bias may be unacceptable, and can be corrected by a Metropolis-Hastings style accept/reject step for each iteration of the algorithm (this is known as Metropolis adjusted Langevin algorithm (MALA) [27]) under the condition that the unnormalised density can be evaluated directly, not just the score as for the unadjusted Langevin algorithm (ULA). ULA does not perform this accept/reject stage and is therefore more computationally efficient than MALA, at the cost of introducing discretization bias [12, 13, 20].

For the examples in this paper, the observation operator $\mathcal{H}$ is described by the point-wise evaluation of the PDE solution at a set of observation locations in the domain, $\{x_j\}_{j=1}^{n_y}$. The discrete observation operator, $\mathbf{H}\colon \mathbb{R}^{n_u} \to \mathbb{R}^{n_y}$, maps the FEM coefficients to observations by interpolating the FEM solution onto these observation locations. We also assume an additive Gaussian noise on the observations, which sets up the linear discrete observation model

$$\mathbf{y} = \mathbf{H}\mathbf{u} + \mathbf{r}, \quad \mathbf{r} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}), \tag{14}$$

where $\mathbf{R} \in \mathbb{R}^{n_y \times n_y}$ is the noise covariance matrix. In the context of Bayesian inference, the target is the *posterior* distribution, $p(\mathbf{u}|\mathbf{y}, \theta, \mathbf{b}) = p(\mathbf{u}, \mathbf{y}, \theta, \mathbf{b}) / \int p(\mathbf{u}, \mathbf{y}, \theta, \mathbf{b}) \mathrm{d}\mathbf{u}$, which in general has an intractable normalisation constant due to the integral $\int p(\mathbf{u}, \mathbf{y}, \theta, \mathbf{b}) \mathrm{d}\mathbf{u}$. Since ULA only relies on the score of the target density, samples from the posterior can be generated without having to compute the normalisation constant [50]. For this posterior inference, the joint potential of latent variables and data $\Phi_\theta^y(\mathbf{u}, \mathbf{b}) \coloneqq \Phi_\theta(\mathbf{u}, \mathbf{b}) - \log p(\mathbf{y}|\mathbf{u})$ is used; the invariant distribution of the following Langevin SDE is posterior $p(\mathbf{u}|\mathbf{y}, \theta, \mathbf{b})$

$$\mathrm{d}\mathbf{u}_t = -\nabla_{\mathbf{u}} \Phi_\theta^y(\mathbf{u}_t, \mathbf{b}) \, \mathrm{d}t + \sqrt{2} \, \mathrm{d}\mathbf{B}_t \tag{15}$$

where $(\mathbf{B}_t)_{t \geq 0}$ is a n_u-dimensional Brownian motion.

Preconditioned diffusion

For some problems, the potential may be ill-conditioned requiring small step-sizes for numerical stability, leading to inefficient sampling. This can be alleviated by preconditioning the Langevin dynamics with a symmetric *preconditioner*, $\mathbf{P} \in \mathbb{R}^{d \times d}$ [27]. The preconditioned Langevin SDE is simply,

$$\mathrm{d}\mathbf{u}_t = -\mathbf{P} \nabla_{\mathbf{u}} \Phi_\theta^y(\mathbf{u}_t, \mathbf{b}) \mathrm{d}t + \sqrt{2} \, \mathbf{P}^{1/2} \mathrm{d}\mathbf{B}_t. \tag{16}$$

Preconditioning can improve the stability and convergence of the sampling scheme, whilst leaving the stationary measure invariant [4]. Section 3.1 shows how preconditioning is applied for posterior sampling.

2.3 Interacting Particle Langevin Dynamics

Whilst sections below will detail the IPLA algorithm for forcing estimation and solving the inverse problem, we provide a brief overview here. IPLA simultaneously evolves a set of particles with Langevin dynamics, which are used to estimate the gradient of the marginal likelihood with respect to parameters. A system of N particles, $\{\mathbf{u}_t^{(n)}\}_{n=1}^N$ are simulated from the Langevin dynamics described in (15), with independent Brownian motions $(\mathbf{B}_t^{(n)})_{t \geq 0}$ for $n = 1, \ldots, N$. These particles and parameters are jointly performing gradient flow on the negative joint log likelihood where the parameter process is driven with a scaled Brownian motion $\sqrt{2/N} \mathrm{d}\mathbf{B}_t^{(0)}$. This noise term, which is the main difference from the methods introduced in [34], allows for non-asymptotic analysis, and in [3], it is shown that given Lipschitz and strong convexity assumptions on the potential, the resulting parameter estimate will converge to the true MMLE, and the rate of convergence is derived in terms of the Lipschitz and strong convexity constants, and time-discretization step-size. The method has also shown to be adaptable to the nondifferentiable and underdamped cases, see, e.g. [21, 38]. Sections 3.1 and 3.2 give the resulting interacting particle systems for estimation of the forcing function and solving the inverse problem, respectively.

3 Interacting Particle Langevin Dynamics for Linear Poisson PDE

One example of an elliptic PDE is the Poisson equation, which describes processes such as steady-state heat diffusion. The strong form of the linear Poisson equation with homogeneous diffusivity coefficient $\varkappa$, is given by

$$-\nabla \cdot (\varkappa \nabla u(x)) = f + \epsilon, \quad x \in \Omega \tag{17}$$

$$u(x) = 0, \quad x \in \partial\Omega. \tag{18}$$

In our case we use the exponential mapping to reparameterise the strong form with diffusivity $\varkappa$, by $\theta = \log \varkappa$ to ensure non-negativity. The bilinear form associated with this strong form, $\mathcal{A}_\theta(u, v) = e^\theta \langle \nabla u, \nabla v \rangle$, produces the stiffness matrix $[\mathbf{A}_\theta]_{ij} = e^\theta \langle \nabla \phi_i, \nabla \phi_j \rangle$. Since the bilinear form is symmetric, the stiffness matrix is symmetric positive definite, and the linear system, $\mathbf{A}_\theta \mathbf{u} = \mathbf{b}$, will have a unique solution.

The joint likelihood model, $p(\mathbf{y}, \mathbf{u}|\mathbf{b}, \theta)$, factorises as the product of the following Gaussian distributions

$$p(\mathbf{u}|\theta, \mathbf{b}) = \mathcal{N}\left(\mathbf{A}_\theta^{-1}\mathbf{b}, \mathbf{A}_\theta^{-1}\mathbf{G}\mathbf{A}_\theta^{-\top}\right), \quad p(\mathbf{y}|\mathbf{u}) = \mathcal{N}\left(\mathbf{H}\mathbf{u}, \mathbf{R}\right), \tag{19}$$

which defines the potential $\Phi_\theta^{\mathbf{y}}(\mathbf{u}, \mathbf{b})$, which can be used for posterior sampling or for MMLE of either $\mathbf{b}$, θ using IPLA.

To further enable MMAP estimation of the forcing $\mathbf{b}$, and the parameter θ, we require prior distributions over these quantities. We will assume independence $p(\mathbf{b}, \theta) = p(\mathbf{b})p(\theta)$, as we will be estimating these individually. The discretised prior distribution over the forcing coefficients, $\mathbf{b}$, can be induced from a GP prior over the forcing function. Given $f \sim \mathcal{GP}(\mu_f, k_f)$, we have $[\boldsymbol{\mu}]_i = \langle \phi_i, \mu_f \rangle$, $[\boldsymbol{\Sigma}]_{ij} = \langle \phi_i, \langle k_f(\cdot, \cdot), \phi_j \rangle \rangle$. For the parameter θ we assume a Gaussian prior, which is equivalent to a log-normal prior over the diffusivity, but with the added constraint of non-negativity when performing gradient update steps. The prior distributions are as follows

$$p(\mathbf{b}) = \mathcal{N}\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}\right), \quad p(\theta) = \mathcal{N}\left(\mu_\theta, \sigma_\theta^2\right). \tag{20}$$

We have now defined each component of the full joint probabilistic model $p(\mathbf{y}, \mathbf{z}, \mathbf{u}, \mathbf{b}, \theta) = p(\mathbf{y}|\mathbf{u})p(\mathbf{u}|\mathbf{b}, \theta)p(\mathbf{b})p(\theta)$, which we use to define the joint potential for the MMAP estimation, using the potential in (15) for posterior sampling, and the prior distributions $p(\mathbf{b}), p(\theta)$:

$$\Psi_\theta^y(\mathbf{u}, \mathbf{b}) := \Phi_\theta^y(\mathbf{u}, \mathbf{b}) - \log p(\mathbf{b}) - \log p(\theta). \tag{21}$$

Using the forcing estimation as an example to illustrate the standard IPLA, we present the interacting Langevin SDE for estimation of $\mathbf{b}$, with θ fixed between iterations. For a system of N particles, $\{\mathbf{u}_t^{(n)}\}_{n=1}^N$, the interacting Langevin SDE has the following form

$$\mathrm{d}\mathbf{b}_t = -\frac{1}{N}\sum_{n=1}^N \nabla_{\mathbf{b}} \Psi_\theta^y(\mathbf{u}_t^{(n)}, \mathbf{b}_t)\mathrm{d}t + \sqrt{\frac{2}{N}}\,\mathrm{d}\mathbf{B}_t^{(0)} \tag{22}$$

$$\mathrm{d}\mathbf{u}_t^{(n)} = -\nabla_{\mathbf{u}} \Psi_\theta^y(\mathbf{u}_t^{(n)}, \mathbf{b}_t)\mathrm{d}t + \sqrt{2}\,\mathrm{d}\mathbf{B}_t^{(n)}, \quad \forall n \in \{1, \ldots, N\}, \tag{23}$$

where $(\mathbf{B}_t^{(n)})_{t\geq 0}$ is an independent Brownian motion for each n. This can be discretised to form the update steps, which are given below for the k^{th} iteration

$$\mathbf{b}_{k+1} = \mathbf{b}_k - \frac{\gamma}{N}\sum_{n=1}^N \nabla_{\mathbf{b}} \Psi_\theta^y(\mathbf{u}_k^{(n)}, \mathbf{b}_k) + \sqrt{\frac{2\gamma}{N}}\,\boldsymbol{\zeta}_{k+1}^{(0)} \tag{24}$$

$$\mathbf{u}_{k+1}^{(n)} = \mathbf{u}_k^{(n)} - \gamma \nabla_{\mathbf{u}} \Psi_\theta^y(\mathbf{u}_k^{(n)}, \mathbf{b}_k) + \sqrt{2\gamma}\,\boldsymbol{\zeta}_{k+1}^{(n)}, \quad \forall n \in \{1, \ldots, N\}, \tag{25}$$

where for each $n \in \{1, \ldots, N\}$, $\{\boldsymbol{\zeta}_k^{(n)}\}_{k\in\mathbb{N}}$ is a sequence of i.i.d. standard Gaussian random variables. This can simply be adapted to MMAP estimation of the diffusivity, by instead fixing $\mathbf{b}$, and replacing (22) to evolve θ with the gradient of the joint potential $\nabla_\theta \Psi_\theta^y(\mathbf{u}, \mathbf{b})$. In the following two sections we derive the particular forms of the IPLA update steps for both the forcing estimation and solving the inverse problem, including preconditioning.

3.1 Forcing Estimation for Linear Poisson PDE

Here, we derive the IPLA update steps for estimating the forcing $\mathbf{b}$, while taking the diffusivity as known and constant. We also apply a fixed preconditioner to this potential to improve stability and sample efficiency. The preconditioners for the latent sampling $\mathbf{P}_{\mathbf{u}}$, and for parameter gradient descent $\mathbf{P}_{\mathbf{b}}$ are set to the inverse of the block diagonal Hessian matrices (see Equation (33)) as follows:

$$\mathbf{P}_{\mathbf{b}} = \left[\mathbf{G}^{-1} + \boldsymbol{\Sigma}^{-1}\right]^{-1}, \quad \mathbf{P}_{\mathbf{u}} = \left[\mathbf{A}_\theta^\top \mathbf{G}^{-1}\mathbf{A}_\theta + \mathbf{H}^\top \mathbf{R}^{-1}\mathbf{H}\right]^{-1}, \tag{26}$$

which individually preconditions both the posterior sampling and the parameter gradient descent with their exact Hessians, but ignores their interaction (off-diagonal terms of the full block Hessian). With this choice of preconditioning the forcing estimation simplifies into the following update steps

$$\mathbf{b}_{k+1} = (1-\gamma)\mathbf{b}_k + \gamma\mathbf{P}_{\mathbf{b}}\left(\mathbf{G}^{-1}\mathbf{A}_\theta\left(\frac{1}{N}\sum_{n=1}^{N}\mathbf{u}_k^{(n)}\right) + \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}\right) + \sqrt{\frac{2\gamma}{N}}\mathbf{P}_{\mathbf{b}}^{1/2}\boldsymbol{\zeta}_{k+1}^{(0)}, \tag{27}$$

$$\mathbf{u}_{k+1}^{(n)} = (1-\gamma)\mathbf{u}_k^{(n)} + \gamma\mathbf{P}_{\mathbf{u}}\left(\mathbf{A}_\theta^\top\mathbf{G}^{-1}\mathbf{b}_k + \mathbf{H}^\top\mathbf{R}^{-1}\mathbf{y}\right) + \sqrt{2\gamma}\mathbf{P}_{\mathbf{u}}^{1/2}\boldsymbol{\zeta}_{k+1}^{(n)}, \tag{28}$$

for all $n \in \{1,\dots,N\}$, with $\{\boldsymbol{\zeta}_k^{(n)}\}_{n=0}^N$ as i.i.d. standard n_u-dimensional Gaussians.

3.2 Solving the Inverse Problem for Linear Poisson PDE

Here, we derive the update steps for estimating the diffusivity, given a known forcing function. Preconditioning is applied at each time-step to the particle Langevin dynamics, based on the current value of θ_k, where $\mathbf{P}_{\mathbf{u}}[\theta_k] = \left[\mathbf{A}_{\theta_k}^\top\mathbf{G}^{-1}\mathbf{A}_{\theta_k} + \mathbf{H}^\top\mathbf{R}^{-1}\mathbf{H}\right]^{-1}$. Writing out the IPLA update steps for this problem, we have

$$\theta_{k+1} = \theta_k - \gamma\left(n_u + \frac{\theta_k - \mu_\theta}{\sigma_\theta^2}\right) - \frac{\gamma}{N}\sum_{n=1}^{N}\left(\mathbf{A}_{\theta_k}\mathbf{u}_k^{(n)} - \mathbf{b}\right)^\top\mathbf{G}^{-1}\mathbf{A}_{\theta_k}\mathbf{u}_k^{(n)} + \sqrt{\frac{2\gamma}{N}}\,\boldsymbol{\zeta}_{k+1}^{(0)} \tag{29}$$

$$\mathbf{u}_{k+1}^{(n)} = (1-\gamma)\mathbf{u}_k^{(n)} + \gamma\mathbf{P}_{\mathbf{u}}[\theta_k]\left(\mathbf{A}_{\theta_k}^\top\mathbf{G}^{-1}\mathbf{b} + \mathbf{H}^\top\mathbf{R}^{-1}\mathbf{y}\right) + \sqrt{2\gamma}\,\mathbf{P}_{\mathbf{u}}[\theta_k]^{1/2}\boldsymbol{\zeta}_{k+1}^{(n)} \tag{30}$$

for all $n \in \{1,\dots,N\}$, with $\boldsymbol{\zeta}_k^{(0)} \sim \mathcal{N}(0,1)$, and $\{\boldsymbol{\zeta}_k^{(n)}\}_{n=1}^N$ as i.i.d. standard n_u-dimensional Gaussians. The factorisation $\mathbf{A}_\theta = e^\theta\mathbf{A}$ allows us to compute the gradient $\nabla_\theta\Psi_\theta^y$ for (29) analytically. For problems without such a factorisation, we could use automatic differentiation to compute $\frac{\partial}{\partial\theta}\mathbf{A}_\theta$ for use in the gradient update steps.

4 Convergence analysis

In this section, we provide convergence analysis of some of the methods developed in this paper. The IPLA method is guaranteed to convergence if the joint potential is strongly convex and globally gradient-Lipschitz continuous. We show below that for the example of elliptic PDE forcing estimation, we can *prove* that these conditions hold and, as a result, the IPLA algorithm will converge to the MMAP estimate.

4.1 Convergence of elliptic PDE forcing estimate to MMAP estimator

Recall that we consider the setting in section 3.1 and in this case, we have the potential $\Psi_\theta^y(\mathbf{u},\mathbf{b})$ for the joint system $\mathbf{v} = (\mathbf{u},\mathbf{b})$ conditioned on the data $\mathbf{y}$. We can write the gradients of the potential w.r.t. the solution, $\mathbf{u}$, and parameters $\mathbf{b}$, are as follows

$$\nabla_{\mathbf{u}}\Psi_\theta^y(\mathbf{u},\mathbf{b}) = \mathbf{A}_\theta^\top\mathbf{G}^{-1}\left(\mathbf{A}_\theta\mathbf{u} - \mathbf{b}\right) + \mathbf{H}^\top\mathbf{R}^{-1}\left(\mathbf{H}\mathbf{u} - \mathbf{y}\right), \tag{31}$$

$$\nabla_{\mathbf{b}}\Psi_\theta^y(\mathbf{u},\mathbf{b}) = -\mathbf{G}^{-1}\left(\mathbf{A}_\theta\mathbf{u} - \mathbf{b}\right) + \boldsymbol{\Sigma}^{-1}(\mathbf{b} - \boldsymbol{\mu}). \tag{32}$$

For this problem we have a quadratic potential, and so the strong convexity and Lipschitz constants are determined by the minimum and maximum eigenvalues of the Hessian:

$$\nabla^2\Psi_\theta^y = \begin{bmatrix}\mathbf{A}_\theta^\top\mathbf{G}^{-1}\mathbf{A}_\theta + \mathbf{H}^\top\mathbf{R}^{-1}\mathbf{H} & -\mathbf{A}_\theta^\top\mathbf{G}^{-1} \\ -\mathbf{G}^{-1}\mathbf{A}_\theta & \mathbf{G}^{-1} + \boldsymbol{\Sigma}^{-1}\end{bmatrix}. \tag{33}$$

Provided the Hessian is positive-definite, we have $\mu > 0$, and a unique minimiser. We can indeed prove this result for this particular problem.

Lemma 1. *The potential $\Psi_\theta^y(\mathbf{u}, \mathbf{b})$ for the linear-Poisson equation is μ-strongly convex in $(\mathbf{u}, \mathbf{b})$, with $\mu = \lambda_{\min}(\nabla^2 \Psi_\theta^y) > 0$.*

Proof. See Appendix A. □

We also show that without the prior, considering only the MMLE potential Φ_θ^y, we can only guarantee $\mu = \lambda_{\min}(\nabla^2 \Phi_\theta^y) \geq 0$ and therefore is only convex, not *strongly* convex. This motivates the use of the prior on the forcing function to ensure identifiability and guarantee a unique minimiser. For this linear case, we can determine this minimiser analytically and we derive this in Appendix C.

Theorem 1. *Assume $L := \lambda_{\max}(\nabla^2 \Phi_\theta^y) < \infty$. Then, the parameter marginal $(\mathbf{b}_k)_{k\geq 1}$ generated by the IPLA algorithm (24)–(25) with the step-size satisfying $\gamma \leq 2/(\mu + L)$ for estimating the forcing function in the linear Poisson equation converges to the MMAP estimator $\mathbf{b}^\star = \arg\max p(\mathbf{b}|\theta, \mathbf{y})$. In particular, we have the following convergence result*

$$\mathbb{E}\left[\|\mathbf{b}_k - \mathbf{b}^\star\|^2\right]^{1/2} \leq \sqrt{\frac{2n_u}{N\mu}} + (1-\mu\gamma)^k W_2(\nu_0, \tilde{\pi}) + 1.65\frac{L}{\mu}\sqrt{\frac{(N+1)n_u}{N}}\gamma^{1/2}, \tag{34}$$

where ν_0 is the initial distribution of the particle system so that $W_2(\nu_0, \tilde{\pi}) < \infty$ where $\tilde{\pi}$ is given in the proof, μ is the strong convexity constant given in Lemma 1.

Proof. See Appendix B. □

We experimentally confirm the result in Theorem 1, by estimating the order of convergence of the IPLA estimate to the MMAP estimate in section 5.1. A few remarks to fully unpack this result are in order.

Remark 1. *Given Theorem 1, we can obtain complexity guarantees for our method to estimate the forcing. Let $\kappa = L/\mu$ denote the condition number. We start by requiring $N = \mathcal{O}(\varepsilon^{-2} n_u)$ which makes the first term in (34) of order ε, i.e., $\mathcal{O}(\varepsilon)$. Next, we set $\gamma = \mathcal{O}(\varepsilon^2 n_u^{-1} \kappa^{-2})$ which makes the last term in (34) of order $\mathcal{O}(\varepsilon)$. Finally, setting $k \geq \widetilde{\mathcal{O}}(n_u \kappa^2 \varepsilon^{-2})$ provides[1] an error so that $\mathbb{E}\left[\|\mathbf{b}_k - \mathbf{b}^\star\|^2\right]^{1/2} \leq \varepsilon$.* ◇

Here, we note the linear dependence of the complexity on the latent dimension n_u, that is for lower latent dimension (i.e. for coarser FEM discretisation) the complexity will reduce, which may appear counter-intuitive. This is a result of bounding the error between the MMAP estimate $\mathbf{b}^\star$ and the IPLA estimate $\mathbf{b}_k$; with n_u too small, the resulting MMAP estimate will not necessarily be close to the true forcing $\mathbf{b}$ due to the large FEM discretisation error incurred. We point readers towards the works [33, 39], which provide error bounds for statFEM between the true solution and the statFEM distributions, in terms of the FEM discretisation.

Remark 2. *Remark 1 makes it clear that the number of iterations k to obtain ε-accuracy is strongly influenced by κ. It is thus crucial to control this number for ill-conditioned problems, as in our case. The algorithm in (27)–(28) is preconditioned and this can potentially improve our bound. Specifically, with a transformation of variables $\mathbf{w} := \mathbf{P}^{-1/2}\mathbf{v}$, it can be shown via Itô's formula that an equivalent SDE, with potential $\Psi_{\theta,\mathbf{P}}^y(\mathbf{w}) := \Psi_\theta^y(\mathbf{v}) = \Psi^y(\mathbf{P}^{1/2}\mathbf{w})$ has the same invariant measure [4]. We can then analyse the Lipschitz and strong convexity constants in the transformed coordinate system, using the same method as in section 4.1 now that the SDE is of the same form. The Hessian for the transformed potential $\nabla_{\mathbf{w}}^2 \Psi_{\theta,\mathbf{P}}^y(\mathbf{w}) = \mathbf{P}^{1/2}\nabla_{\mathbf{v}}^2 \Psi_\theta^y(\mathbf{v})\mathbf{P}^{1/2}$, and so the constants relating to the preconditioned system are*

$$\mu_{\mathbf{P}} = \lambda_{\min}\left(\mathbf{P}^{1/2}\nabla^2\Psi_\theta^y\mathbf{P}^{1/2}\right), \quad L_{\mathbf{P}} = \lambda_{\max}\left(\mathbf{P}^{1/2}\nabla^2\Psi_\theta^y\mathbf{P}^{1/2}\right). \tag{35}$$

The bound in Theorem 1 then applies directly with the constants $\mu_{\mathbf{P}}$ and $L_{\mathbf{P}}$. Let $\kappa_{\mathbf{P}} = L_{\mathbf{P}}/\mu_{\mathbf{P}}$ be the condition number of the preconditioned system. Given Remark 1 and assuming that $\kappa_{\mathbf{P}} \ll \kappa$, preconditioning has two clear effects: First, the step-size $\gamma = \mathcal{O}(\varepsilon^2 n_u^{-1} \kappa_{\mathbf{P}}^{-2})$ can now be bigger (within the restrictions provided in Theorem 1). Secondly, to obtain ε-accuracy, we may require significantly fewer steps, i.e., $k = \widetilde{\mathcal{O}}(n_u \kappa_{\mathbf{P}}^2 \varepsilon^{-2})$. ◇

[1] The notation $\widetilde{\mathcal{O}}$ suppresses logarithmic factors.

5 Experimental Results

In this section, we provide comprehensive numerical evidence that the methods we presented above are effective in practice. We consider two examples, with different levels of complexity, to demonstrate the effectiveness of the IPLA algorithm within the setting of inverse problems. Specifically, we consider the following examples:

- **Linear Poisson forcing estimation:** We include this example because both the exact posterior, and the exact MMAP estimator are available analytically due to the linearity of the problem. This allows us to experimentally confirm the convergence rate of our IPLA estimate of the forcing to the ground truth MMAP estimator. We also assess the stability of the sampler with varying time-discretisation, and with and without preconditioning.
- **Linear Poisson inverse problem:** We consider the estimation of an unknown constant diffusivity for the linear Poisson problem, demonstrating the effect of a warm-start on the efficiency of estimating the parameter.

5.1 Forcing estimation for Linear Poisson PDE

Here we consider two example domains, the unit interval, and a 2D disc. In both cases we fix the diffusivity to a constant value of one, i.e. $\varkappa = 1$. For the unit interval the linear Poisson PDE is given by

$$\begin{aligned} -\nabla \cdot (\nabla u(x)) &= f + \epsilon, \quad x \in (0,1), \quad \epsilon \sim \mathcal{GP}(0,k) \\ u(x) &= 0, \quad x \in \{0,1\}, \end{aligned} \tag{36}$$

where $f(x) = 5\sin(6\pi x)$, $k(x,x') = \exp\left(-\frac{\|x-x'\|_2^2}{2*0.1^2}\right)$, and for the forcing prior $f \sim \mathcal{GP}(0,k_f)$, $k_f(x,x') = 4\exp\left(-\frac{\|x-x'\|_2^2}{2*0.1^2}\right)$. For the disc, $\Omega = \left\{\mathbf{x} \in \mathbb{R}^2 \colon \|\mathbf{x}\| < 1\right\}$, we have:

$$\begin{aligned} -\nabla \cdot (\nabla u(\mathbf{x})) &= f + \epsilon, \quad \mathbf{x} \in \Omega, \quad \epsilon \sim \mathcal{GP}(0,k) \\ u(\mathbf{x}) &= 0, \quad \mathbf{x} \in \partial\Omega, \end{aligned} \tag{37}$$

with $f(\mathbf{x}) = 100\exp\left(-\frac{\|\mathbf{x}+\mathbf{m}\|_2^2}{2*0.2^2}\right) + 100\exp\left(-\frac{\|\mathbf{x}-\mathbf{m}\|_2^2}{2*0.2^2}\right)$, $\mathbf{m} = [0.4, 0.4]^\top$, and $k(\mathbf{x},\mathbf{x}') = \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}'\|_2^2}{2*0.1^2}\right)$. For inference $f \sim \mathcal{GP}(0,k_f)$, $k_f(\mathbf{x},\mathbf{x}') = 100\exp\left(-\frac{\|\mathbf{x}-\mathbf{x}'\|_2^2}{2*0.2^2}\right)$. For the examples in this work we assume the GP hyperparameters, such as kernel length-scale and amplitude are known *a priori*. For the forcing prior hyperparameters, these would be set by the practitioner to reflect the *a priori* knowledge. The GP misspecification hyperparameters would ideally be estimated from data. In this study we do not include such estimation, which could be performed by e.g. extending the MMLE/MMAP to include the kernel hyperparameters, via grid search, or Bayesian optimisation methods; we leave this as a consideration for future work.

Convergence results

The convergence of the estimated forcing to the MMAP estimator is computed by assuming a convergence of the form $\|\mathbf{b}_k^{(N)} - \mathbf{b}^\star\|_2 = CN^{-p}$, where $\mathbf{b}_k^{(N)}$ is the parameter estimate after k iterations of preconditioned IPLA (27)–(28) using N particles. The *rate* of convergence, C will depend on the problem and is not the target of this analysis while the *order* of convergence, p is of interest and can be numerically approximated. The logarithm of the error is given by $\log\left(\|\mathbf{b}_k^{(N)} - \mathbf{b}^\star\|\right) = \log C - p\log N$. For example, consider the case where we double N and take the log-ratio of the errors and take the difference of the logarithms: $\log_2\left(\|\mathbf{b}_k^{(N)} - \mathbf{b}^\star\|\right) - \log_2\left(\|\mathbf{b}_k^{(2N)} - \mathbf{b}^\star\|\right) = p$. Running the algorithm multiple times on randomised datasets gives us an average estimate of $\|\mathbf{b}_k^{(N)} - \mathbf{b}^\star\|$ for a particular value of N. We also compute the L^2-error between

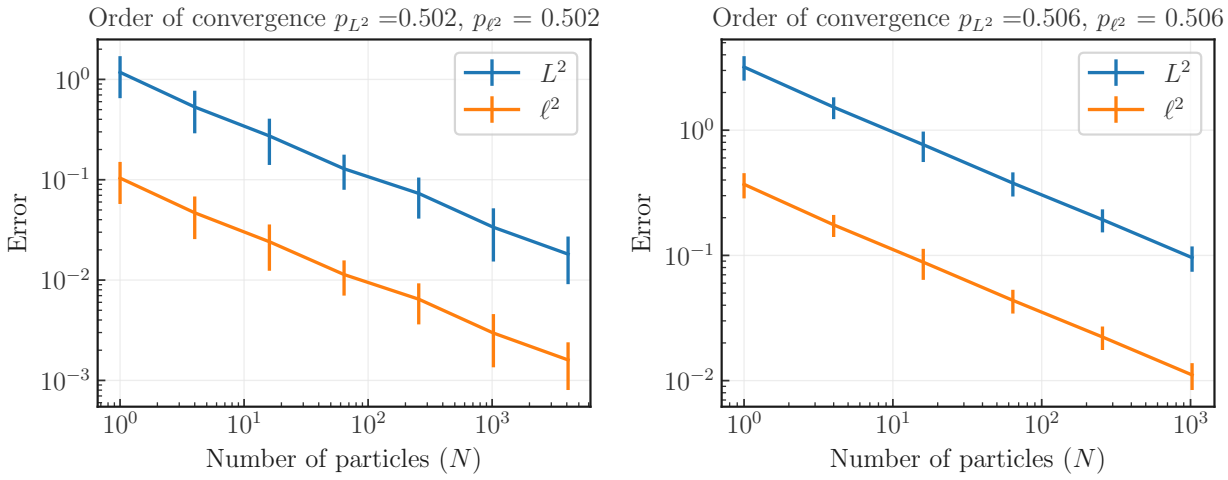

Figure 1: Experimental validation of the order of convergence derived in Theorem 1 for geometries (a) 1D Poisson and (b) Poisson on 2D disc. Theorem 1 proves an error bound on the MMAP estimator, we compute this error for increasing number of particles N, showing $N^{-1/2}$ order of convergence. We refer to $\|\mathbf{b}_k - \mathbf{b}^\star\|$, $\|f_k - f^\star\|$ as the ℓ^2-error and L^2-error, respectively. Error bars computed using 10 Monte Carlo simulations.

the FEM interpolated functions $\|f_k - f^\star\|$, as a comparison. The order of convergence for two examples (1D Poisson on unit interval, 2D Poisson on disc) can be seen in Figure 1 and are estimated to be $(0.502, 0.506)$ are close to $1/2$, which is the theoretical order of convergence derived in Theorem 1. We should also see the particle estimate of the posterior converge to the true posterior, $p(\mathbf{u}|\mathbf{y}, \theta, \mathbf{b}^\star)$; to assess this we compute the variance of the empirical estimate and compare this against the true posterior variance for increasing numbers of particles, which we visualise in Figure 2 for the 1D Poisson example (36).

Robustness

Ideally, the sampling should be numerically stable and resulting inference independent of the degree of mesh refinement for the problem; this corresponds to keeping the dimension of the data n_y fixed, whilst increasing the latent dimension n_u. We have found that increasing the dimension of the latent space makes the problem more ill-posed, particularly when specifying a covariance length-scale that is larger than the FEM node separation making the condition number of the forcing error covariance very large. To demonstrate this, we ran the standard ULA algorithm, without preconditioning for the 1D linear Poisson problem (36), where for $\Omega = [0, 1]$ we specify nodes $x_i = \frac{i}{n_u - 1}, i \in \{0, \ldots, n_u - 1\}$, with piecewise linear basis functions, giving $\mathbf{u} \in \mathbb{R}^{n_u}$. Increasing the dimension n_u of the state-space and finding the maximum step-size before the sampler diverges gives an indication of the numerical stability of the sampler, results for this are shown in Table 1. Preconditioning the Langevin sampling mitigates this problem. To demonstrate the robustness of

Table 1: Maximum step-sizes for a stable sampler for IPLA *without* preconditioning, for increasing mesh refinement and different statFEM prior length-scales, ℓ, where $k(x, x') = \sigma \exp(-\frac{\|x - x'\|^2}{2\ell^2})$. Fixed dataset size $n_y = 16$ used, for 1D Poisson on unit interval domain.

n_u	5	10	20	30	50
max γ ($\ell = 0.01$)	$10^{-4.78}$	$10^{-5.64}$	$10^{-6.87}$	$10^{-7.58}$	$10^{-8.61}$
max γ ($\ell = 0.1$)	$10^{-4.78}$	$10^{-6.94}$	$< 10^{-10}$	$< 10^{-10}$	$< 10^{-10}$

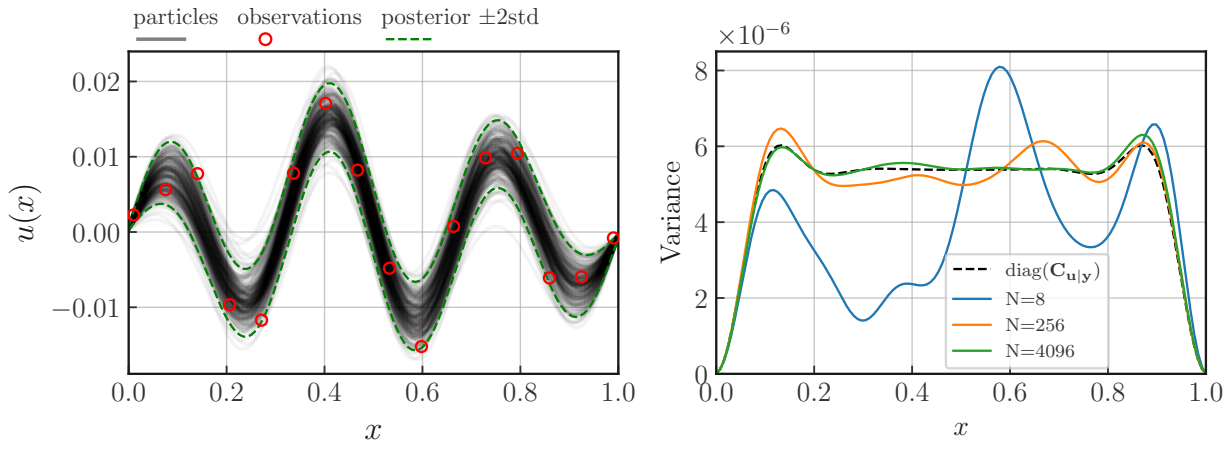

Figure 2: (*left*) True posterior $p(\mathbf{u}|\mathbf{y}, \theta, \mathbf{b}^\star) = \mathcal{N}(\mathbf{m}_{\mathbf{u}|\mathbf{y}}, \mathbf{C}_{\mathbf{u}|\mathbf{y}})$, compared to particle approximated posterior $p(\mathbf{u}|\mathbf{y}, \mathbf{b}_k) = \frac{1}{N}\sum_{n=1}^{N} \delta_{\mathbf{u}_k^{(n)}}$, for $N = 256$. (*right*) Variance of true posterior vs approximations for increasing number of particles.

Table 2: Mesh-refinement results for preconditioned IPLA: Condition numbers for standard and preconditioned potential for increasing n_u.

n_u	32	64	128	256	512
κ	1.14×10^{10}	2.30×10^{10}	4.60×10^{10}	9.20×10^{10}	2.83×10^{11}
$\kappa_{\mathbf{P}}$	65.6	65.6	65.5	65.3	65.0

preconditioning to mesh-refinement, we calculate the condition number for the standard potential, κ, and the preconditioned potential, $\kappa_{\mathbf{P}}$, for increasing latent dimension. Table 2 shows the condition number increasing significantly for the standard potential, whilst the preconditioned case the condition number is small, and constant with respect to the latent dimension.

5.2 Estimating Diffusivity Parameter for Linear Poisson PDE

For the inverse problem of estimating the diffusivity, we use the 1D Poisson equation:

$$\begin{aligned} -\nabla \cdot (e^{\theta} \nabla u) &= f + \epsilon, \quad x \in (0,1), \\ u(x) &= 0, \quad x \in \{0,1\}. \end{aligned} \tag{38}$$

where data is generated from $f(x) = 20\sin(4\pi x)$ and with $\varkappa = 1$. For inference we use $\epsilon \sim \mathcal{GP}(0, k)$, $k(x, x') = 3\exp\left(-\frac{\|x-x'\|_2^2}{2*0.02^2}\right)$, with a prior $p(\theta) = \mathcal{N}(1.5, 0.75^2)$. Estimation results are shown in Figure 3, showing convergence of the parameter estimate to the true value. In this particular experiment, we use a warm-start strategy for $\mathbf{u}$ variables and found that without this warm-start, the algorithm might take a long time to converge. The reason for this can be seen from the gradient of the potential which reads as

$$\nabla_{\mathbf{u}} \Psi_\theta^y(\mathbf{u}, \mathbf{b}) = \mathbf{A}_\theta^\top \mathbf{G}^{-1} \left(\mathbf{A}_\theta \mathbf{u} - \mathbf{b}\right) + \mathbf{H}^\top \mathbf{R}^{-1} \left(\mathbf{H}\mathbf{u} - \mathbf{y}\right).$$

This potential is ill-behaved in θ-dimension with growing Lipschitz coefficients, creating instability. However, observing that, for an initial (fixed) point θ_0, we have the strong convexity

$$\left\langle \mathbf{u} - \mathbf{u}', \nabla_{\mathbf{u}} \Psi_{\theta_0}^y(\mathbf{u}, \mathbf{b}) - \nabla_{\mathbf{u}} \Psi_{\theta_0}^y(\mathbf{u}', \mathbf{b}) \right\rangle \geq \mu \|\mathbf{u} - \mathbf{u}'\|^2,$$

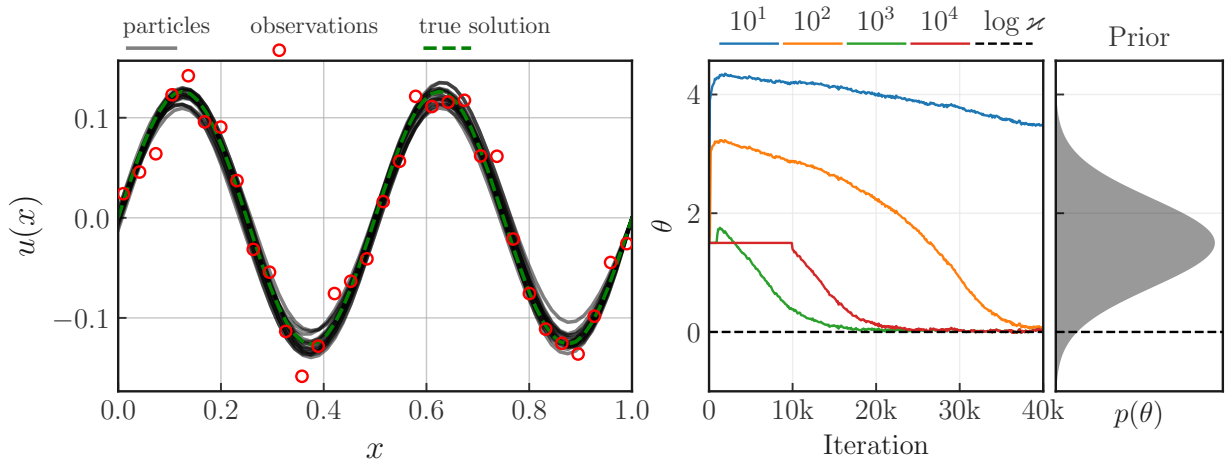

Figure 3: (*left*) Particles $\{\mathbf{u}_k^{(n)}\}_{n=1}^N$ plotted in black, alongside the solution $u(x)$ (dashed green) that generated the data $\mathbf{y}$ (red circles), for warm-start length 10^3, step-size $\gamma = 0.001$. (*center*) Comparison of parameter traces, for different lengths of warm-start compared to true value $\varkappa = 1$, initalised at $\theta_0 = 1.5$. (*right*) Prior distribution, $p(\theta) = \mathcal{N}(1.5, 0.75^2)$.

and Lipschitz continuity

$$\left\|\nabla_{\mathbf{u}} \Psi^y_{\theta_0}(\mathbf{u}, \mathbf{b}) - \nabla_{\mathbf{u}} \Psi^y_{\theta_0}(\mathbf{u}', \mathbf{b})\right\| \leq L \|\mathbf{u} - \mathbf{u}'\|,$$

where

$$\mu = \lambda_{\min}\left(\mathbf{A}_{\theta_0}^\top \mathbf{G}^{-1} \mathbf{A}_{\theta_0} + \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H}\right), \quad L = \lambda_{\max}\left(\mathbf{A}_{\theta_0}^\top \mathbf{G}^{-1} \mathbf{A}_{\theta_0} + \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H}\right).$$

We can run the diffusion on $\mathbf{u}$ with fixed θ_0 which will exhibit exponential convergence. Initialising $\mathbf{u}$ variables at stationarity for θ_0 and running the system jointly produces a stable behaviour experimentally, which might be an interesting starting point of an analysis for this case. We refer to the iteration of the particle update (30) while fixing θ_0 constant a 'warm-start', and the number of iterations for which this is run the 'warm-start length'. The particles are initialised at zero and the diffusivity parameter at some value larger than the truth. With shorter warm-starts, the parameter estimate increases rapidly from the initial value away from the true value because the particles have not had sufficient time to move away from zero, hence the parameter updates believe the system is highly diffusive. The second longest warm-start provides the quickest converge to the true diffusivity, with the longest spending too long in the warm-start stage. Figure 3 shows the parameter traces for these different lengths of warm-start, alongside the prior over the parameter θ.

6 Nonlinear Extension

As we have seen in section 2.1, if the PDE is defined by a linear differential operator the prior distribution induced by the GP noise is also a Gaussian. This is not the case in general. A nonlinear differential operator will induce a (typically non-analytic) non-Gaussian prior over the FEM coefficients. In this section we derive the statFEM prior for a nonlinear PDE, before deriving the potential and approximations for a nonlinear Poisson PDE, before demonstrating the forcing estimation results.

6.1 Statistical Finite Elements for Nonlinear PDEs

A stochastic PDE with the nonlinear differential operator $\mathcal{F}$ and forcing f

$$\mathcal{F}(u) = f + \epsilon, \quad \epsilon \sim \mathcal{GP}(0, k(\cdot, \cdot)), \tag{39}$$

can be discretised via the statFEM method giving a system of nonlinear algebraic equations $\mathbf{F}(\mathbf{u}) = \mathbf{b} + \mathbf{e}, \quad \mathbf{e} \sim \mathcal{N}(\mathbf{0}, \mathbf{G})$ (see Appendix D for derivation). The induced prior distribution can be found via a transformation of variables [7], from distribution $\mathbf{E} \sim f_{\mathbf{E}}(\mathbf{e})$ to $\mathbf{U} \sim f_{\mathbf{U}}(\mathbf{u})$ where $\mathbf{u} = g(\mathbf{e})$,

$$f_{\mathbf{U}}(\mathbf{u}) = \det \left| \frac{\partial g^{-1}(\mathbf{u})}{\partial \mathbf{u}} \right| f_{\mathbf{E}}(g^{-1}(\mathbf{u})). \tag{40}$$

Using this transformation of variables with $f_{\mathbf{E}} = \mathcal{N}(\mathbf{0}, \mathbf{G})$ and $g^{-1}(\mathbf{u}) = \mathbf{F}(\mathbf{u}) - \mathbf{b}$, we can see how the resulting density of the solution depends on the nonlinear differential operator and its Jacobian:

$$p(\mathbf{u}|\mathbf{b}) = \det |\mathbf{J}(\mathbf{u})| \frac{1}{Z} \exp \left\{ -\frac{1}{2} \left(\mathbf{F}(\mathbf{u}) - \mathbf{b} \right)^\top \mathbf{G}^{-1} \left(\mathbf{F}(\mathbf{u}) - \mathbf{b} \right) \right\}, \tag{41}$$

where the constant $Z = (2\pi)^{-n_u/2} \det |\mathbf{G}|^{-1/2}$, and the Jacobian of the nonlinear transform $\mathbf{F}$ is denoted $\mathbf{J}(\mathbf{u}) = \frac{\partial \mathbf{F}}{\partial \mathbf{u}}\big|_{\mathbf{u}}$. This prior, in general, is non-Gaussian due to the Jacobian determinant term.

6.2 Forcing Estimation for Nonlinear Poisson PDE

The strong form of a nonlinear Poisson equation is shown in Equation (42), where we introduce the nonlinearity through a solution-dependent diffusivity, $q(u)$:

$$-\nabla \cdot (q(u) \nabla u) = f + \epsilon. \tag{42}$$

We proceed by multiplying with a test function and integrating over the domain, before applying integration by parts to convert this to the weak form

$$\int_\Omega q(u) \nabla u \cdot \nabla v \mathrm{d}x = \int_\Omega (f + \epsilon)\, v \mathrm{d}x. \tag{43}$$

For full details see Appendix D. With a basis function expansion $u_h = \sum_{i=1}^{n_u} \hat{u}_i \phi_i$, this is rewritten in vector form

$$\mathbf{F}(\mathbf{u}) = \mathbf{b} + \mathbf{e}, \quad \mathbf{e} \sim \mathcal{N}(\mathbf{0}, \mathbf{G}), \tag{44}$$

where we have $\mathbf{u} = [\hat{u}_1, \hat{u}_2, \ldots, \hat{u}_{n_u}]^\top$, $[\mathbf{F}(\mathbf{u})]_i = \langle q(u_h) \nabla u_h, \nabla \phi_i \rangle$, $[\mathbf{b}]_i = \langle f, \phi_i \rangle$, and $[\mathbf{G}]_{ij} = \langle \phi_i, \langle k(\cdot,\cdot), \phi_j \rangle \rangle$. The prior potential we define as $\Phi(\mathbf{u}, \mathbf{b}) \coloneqq -\log p(\mathbf{u}|\mathbf{b})$, and posterior potential $\Phi^y(\mathbf{u}, \mathbf{b}) \coloneqq \Phi(\mathbf{u}, \mathbf{b}) - \log p(\mathbf{y}|\mathbf{u})$ using the same observation model as (2). Finally, in the same manner as the linear Poisson case we use a GP prior for the forcing function, and define the joint potential as $\Psi^y(\mathbf{u}, \mathbf{b}) \coloneqq \Phi^y(\mathbf{u}, \mathbf{b}) - \log p(\mathbf{b})$.

In this nonlinear case the computation of the gradient of the prior potential with respect to $\mathbf{u}$, $\nabla_{\mathbf{u}} \Phi(\mathbf{u}, \mathbf{b})$ is intractable due to the log-determinant Jacobian term in the score of the nonlinear statFEM prior, so we turn to approximations of this distribution to make progress. Approximating this distribution with a Gaussian alleviates the problem of intractability, providing a linear score computed using the approximate mean and covariance. We present here the algorithm for sampling based on a Gaussian approximation of the nonlinear statFEM prior $p(\mathbf{u}|\mathbf{b}_k) \approx \mathcal{N}(\mathbf{m}_k, \mathbf{C}_k)$:

$$\mathbf{b}_{k+1} = (1-\gamma)\mathbf{b}_k + \gamma \mathbf{P}_{\mathbf{b}} \left(\mathbf{G}^{-1} \left(\frac{1}{N} \sum_{n=1}^{N} \mathbf{F}(\mathbf{u}_k^{(n)}) \right) + \mathbf{\Sigma}^{-1} \boldsymbol{\mu} \right) + \sqrt{\frac{2\gamma}{N}} \mathbf{P}_{\mathbf{b}}^{1/2} \zeta_{k+1}^{(0)} \tag{45}$$

$$\mathbf{u}_{k+1}^{(n)} = (1-\gamma)\mathbf{u}_k^{(n)} + \gamma \mathbf{P}_{\mathbf{u}} \left(\mathbf{C}_k^{-1} \mathbf{m}_k + \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{y} \right) + \sqrt{2\gamma} \mathbf{P}_{\mathbf{u}}^{1/2} \zeta_{k+1}^{(n)}, \quad \forall n \in \{1, \ldots, N\} \tag{46}$$

$$\text{with} \quad \mathbf{P}_{\mathbf{b}} = \left[\mathbf{G}^{-1} + \mathbf{\Sigma}^{-1} \right]^{-1}, \quad \mathbf{P}_{\mathbf{u}} = \left[\mathbf{C}_k^{-1} + \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H} \right]^{-1}. \tag{47}$$

Note the gradient with respect to the forcing can be computed exactly because the potential is quadratic in $\mathbf{b}$, i.e. $\nabla_{\mathbf{b}} \Psi^y(\mathbf{u}, \mathbf{b}) = \mathbf{G}^{-1} (\mathbf{F}(\mathbf{u}) - \mathbf{b})$. Simple approaches to approximating the distribution of nonlinearly

transformed Gaussian random variables [29] include a first-order Taylor approximation, the unscented transform, and an empirical approximation to the first two moments with a sample based approach (See Table 3). The update steps (45)–(46) require the computation of the preconditioner at each iteration as the approximation to $p(\mathbf{u}|\mathbf{b}_k)$ is updated. This is a computationally expensive step, which requires the matrix inverse $\mathbf{C}_k^{-1}$ requiring order n_u^3 operations, followed by the inverse $\left[\mathbf{C}_k^{-1} + \mathbf{H}^\top \mathbf{R}^{-1}\mathbf{H}\right]^{-1}$, again order n_u^3. We note that for first-order Taylor, we can compute the inverse of covariance matrix efficiently from the Jacobian by noting $\mathbf{C}_k^{-1} = \mathbf{J}(\mathbf{m}_k)^\top \mathbf{G}^{-1}\mathbf{J}(\mathbf{m}_k)$ which is only order n_u^2. For larger-scale problems when this may become infeasible to recompute the preconditioner at each iteration other methods can be used for preconditioning, since for the first-order Taylor method, the preconditioning computation is order n_u^3 but the approximation is order n_u^2, hence fixing this preconditioner for some iterations may have significant impacts on efficiency. For example, a fixed preconditioned could be used, computed at the prior mean, or the preconditioner could be recomputed after a fixed number of steps. These different preconditioning strategies are of interest in future work.

We remark that all these methods require us to solve the nonlinear system, and the first-order Taylor method requires computation of the Jacobian. This procludes the use of the first-order Taylor method for non-differentiable nonlinearities, where we can only evaluate $\mathbf{F}$ and not its Jacobian $\mathbf{J}$; for this case, fixed-point methods could be used to solve $\mathbf{F}(\mathbf{u}) = \mathbf{b}$ and the unscented transform or Monte Carlo methods applied for approximating the covariance. For this example, since we have access to the Jacobian, we apply Newton's method for solving the system. Details of each of these approximations are given in Appendix E.2.

Table 3: Gaussian approximations to nonlinearly transformed Gaussian random variables: first-order Taylor (FOT), unscented transform (UT) and Monte Carlo (MC). FOT requires one solve followed by computation of the Jacobian, $\mathbf{J}(\hat{\mathbf{m}}) = \frac{\partial \mathbf{F}}{\partial \mathbf{u}}|_{\hat{\mathbf{m}}}$. UT requires $2n_u + 1$ solves, details in Appendix E.2. MC requires M solves, where $\tilde{\mathbf{b}}^{(j)} \sim \mathcal{N}(\mathbf{b}, \mathbf{G})$.

	Solve	Mean: $\mathbf{m}$	Covariance: $\mathbf{C}$
FOT	$\mathbf{F}(\hat{\mathbf{m}}) = \mathbf{b}$	$\hat{\mathbf{m}}$	$\mathbf{J}(\hat{\mathbf{m}})^{-1}\mathbf{G}\mathbf{J}(\hat{\mathbf{m}})^{-\top}$
UT $j \in \{-n_u, \ldots, n_u\}$	$\mathbf{F}(\mathbf{u}^{(j)}) = \mathbf{s}^{(j)}$	$\sum_j w_j \mathbf{u}^{(j)}$	$\sum_j \hat{w}_j(\mathbf{u}^{(j)} - \mathbf{m})(\mathbf{u}^{(j)} - \mathbf{m})^\top$
MC $j \in \{1, \ldots, M\}$	$\mathbf{F}(\mathbf{u}^{(j)}) = \tilde{\mathbf{b}}^{(j)}$	$\frac{1}{M}\sum_{j=1}^M \mathbf{u}^{(j)}$	$\frac{1}{M-1}\sum_{j=1}^M (\mathbf{u}^{(j)} - \mathbf{m})(\mathbf{u}^{(j)} - \mathbf{m})^\top$

6.3 Nonlinear Poisson Experimental Results

This example introduces a nonlinearity into the Poisson diffusivity, where we can compare different nonlinear approximation methods. Here, we compare the first-order Taylor, unscented transform and Monte Carlo approximations for approximating nonlinear transforms of Gaussian random variables. For the nonlinear Poisson PDE (42), we introduce a known nonlinearity with a solution-dependent diffusivity $q(x) = 1 + u(x)^2$:

$$\begin{aligned} -\nabla \cdot ((1 + u(x)^2)\nabla u(x)) &= f + \epsilon, \quad x \in (0,1), \quad \epsilon \sim \mathcal{GP}(0, k) \\ u(0) = 0,\ u(1) &= 1, \end{aligned} \tag{48}$$

where $f(x) = 10\sin(2\pi x)$, $k(x, x') = \exp\left(-\frac{\|x-x'\|_2^2}{2*0.02^2}\right)$ for model misspecification, and $f \sim \mathcal{GP}(0, k_f)$, with $k_f(x, x') = 6\exp\left(-\frac{\|x-x'\|_2^2}{2*0.1^2}\right)$ for the forcing prior. We compare the three nonlinear approximations in Table 3, showing parameter estimation and posterior inference results in Figure 4. The first-order Taylor approximation has the smallest L^2-error between the estimated and true forcing, and is considerably less computationally expensive. Using this approximation method we assess the convergence of the forcing estimate to the true forcing with increasing particles (Figure 5).

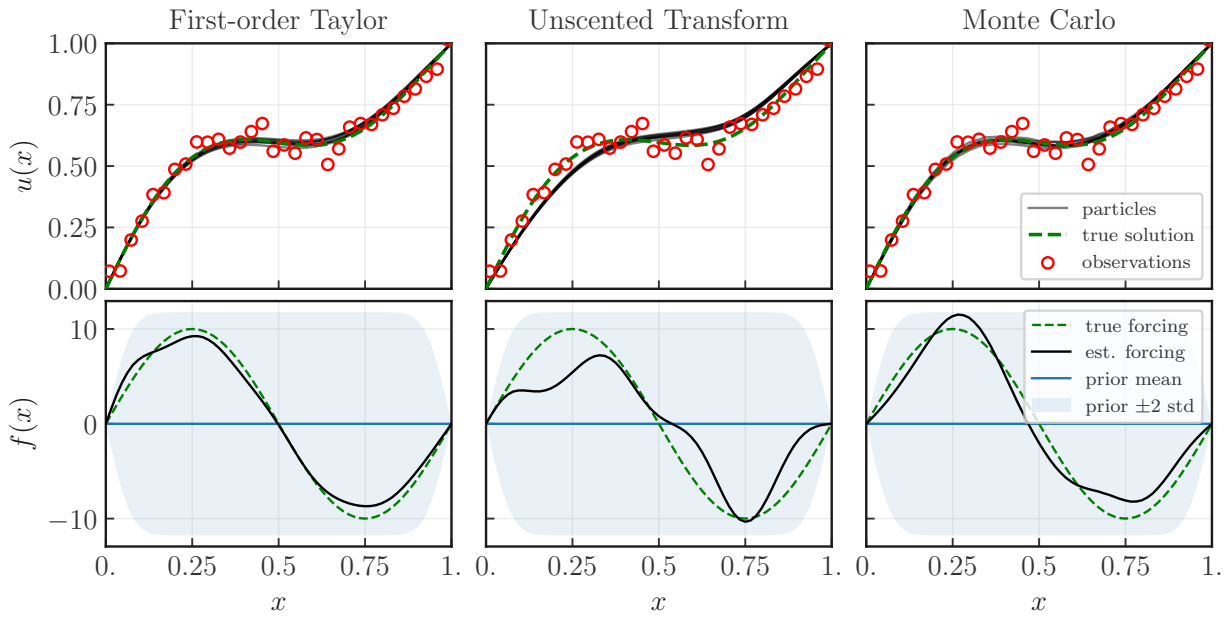

Figure 4: Comparison of forcing estimation results for different Gaussian approximations (See Table 3) to the nonlinear statFEM prior (42). First-order Taylor approximation is the fastest to compute, and provides the best estimate of the true forcing function. Results shown at $k = 10^4$ iterations, with $\gamma = 0.005$ and $N = 4$.

7 Conclusions

In this paper we have posed the problem of estimating PDE parameters as a MMAP problem, constructing statFEM generative model, to which we applied IPLA. Nonasymptotic bounds on the MMAP error have been proved, and experimentally confirmed for both 1D and 2D geometries. The approach in [4] has been extended to a nonlinear statFEM setting, and applied to the nonlinear Poisson PDE. Different methods for constructing Gaussian approximations to the non-Gaussian statFEM prior were considered and tested, with a particular focus on first-order Taylor approach. The effects of a 'warm-start' were explored for the case of solving the inverse problem of estimating diffusivity, showing that a warm-start can improve the convergence of the estimate. With nonasymptotic bounds derived, further research will consider scaling up this approach to work with larger systems, where solving the forward problem is prohibitively expensive — in this case, Langevin dynamics can still be used to sample from the target density without forward solves [4]. Future work can also benefit from extensions to non-log-concave cases arising in nonlinear PDEs using the techniques introduced in [51, 2]. Recent developments in particle-based schemes for accelerating the gradient descent are of interest in future work, such as the underdamped Langevin extension to IPLA [38], the momentum-based approach of [35], and the Stein variational gradient descent method of [43].

Acknowledgements

AGD was supported by Splunk Inc. [G106483] Ph.D scholarship funding. CD and MG were supported by EPSRC grant EP/T000414/1. MG was supported by a Royal Academy of Engineering Research Chair, and Engineering and Physical Sciences Research Council (EPSRC) grants EP/T000414/1, EP/W005816/1, EP/V056441/1, EP/V056522/1, EP/R018413/2, EP/R034710/1, and EP/R004889/1.
The data that support the findings of this study are available from the corresponding author, AGD, upon reasonable request.

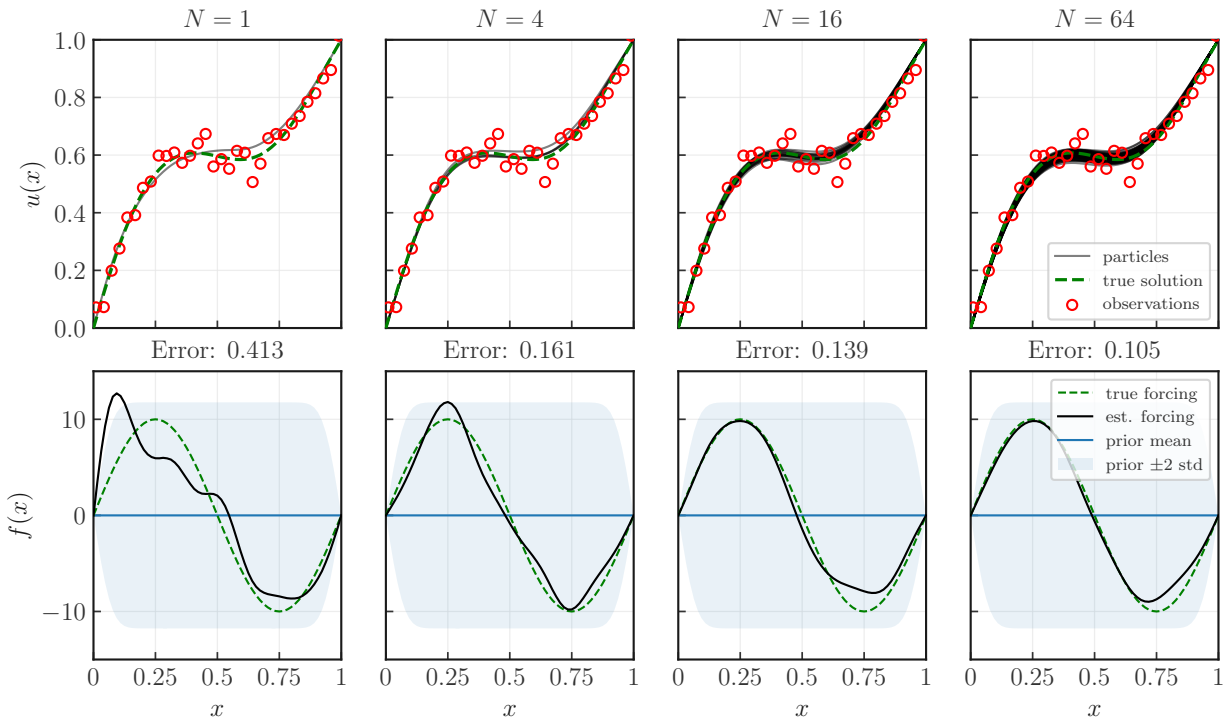

Figure 5: Estimation results for the first-order Taylor approximation to nonlinear statFEM prior for increasing number of particles N. (*above*) Particles $\{\mathbf{u}_k^{(n)}\}_{n=1}^N$ plotted in black, alongside the solution $u(x)$ (dashed green) that generated the data $\mathbf{y}$ (red circles). (*below*) We can see steady improvement of the forcing estimate $f_k(x) = \sum_{i=1}^{n_u} [\mathbf{b}_k]_i \phi_i(x)$, to the true forcing function $f(x) = 10\sin(2\pi x)$. Results shown at $k = 10^4$ iterations, with $\gamma = 0.005$.

References

[1] Henry D. I. Abarbanel, Daniel R. Creveling, Reza Farsian, and Mark Kostuk. Dynamical State and Parameter Estimation. *SIAM Journal on Applied Dynamical Systems*, 8(4):1341–1381, January 2009. ISSN 1536-0040. doi: 10.1137/090749761. 3

[2] O Deniz Akyildiz and Sotirios Sabanis. Nonasymptotic Analysis of Stochastic Gradient Hamiltonian Monte Carlo Under Local Conditions for Nonconvex Optimization. *Journal of Machine Learning Research*, 25(113):1–34, 2024. 16

[3] Ö Deniz Akyildiz, Francesca Romana Crucinio, Mark Girolami, Tim Johnston, and Sotirios Sabanis. Interacting Particle Langevin Algorithm for Maximum Marginal Likelihood Estimation. *ESAIM: Probability and Statistics*, 2025. 1, 3, 6, 23

[4] Ömer Deniz Akyildiz, Connor Duffin, Sotirios Sabanis, and Mark Girolami. Statistical Finite Elements via Langevin Dynamics. *SIAM/ASA Journal on Uncertainty Quantification*, 10(4):1560–1585, December 2022. 1, 2, 3, 6, 9, 16

[5] Yves F Atchadé, Gersende Fort, and Eric Moulines. On Perturbed Proximal Gradient Algorithms. *Journal of Machine Learning Research*, 18(1):310–342, 2017. 2

[6] Juliana Atmadja and Amvrossios C Bagtzoglou. Pollution Source Identification in Heterogeneous Porous Media. *Water Resources Research*, 37(8):2113–2125, 2001. 3

[7] Patrick Billingsley. *Probability and Measure*. John Wiley & Sons., 1995. 14

[8] Stephen P Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge university press, 2004. 21

[9] Susanne C Brenner. *The Mathematical Theory of Finite Element Methods*. Springer, 2008. 24

[10] Rocco Caprio, Juan Kuntz, Samuel Power, and Adam M Johansen. Error Bounds for Particle Gradient Descent, and Extensions of the Log-Sobolev and Talagrand Inequalities. *arXiv preprint arXiv:2403.02004*, 2024. 3

[11] Gilles Celeux and Jean Diebolt. A Stochastic Approximation Type EM Algorithm for the Mixture Problem. *Stochastics: An International Journal of Probability and Stochastic Processes*, 41(1-2):119–134, 1992. 2

[12] Arnak S Dalalyan. Theoretical Guarantees for Approximate Sampling from Smooth and Log-Concave Densities. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, pages 651–676, 2017. 2, 5

[13] Arnak S Dalalyan and Avetik Karagulyan. User-Friendly Guarantees for the Langevin Monte Carlo with Inaccurate Gradient. *Stochastic Processes and their Applications*, 129(12):5278–5311, 2019. 5, 22

[14] Valentin De Bortoli, Alain Durmus, Marcelo Pereyra, and Ana F Vidal. Efficient Stochastic Optimisation by Unadjusted Langevin Monte Carlo. *Statistics and Computing*, 31(3):1–18, 2021. 2

[15] Bernard Delyon, Marc Lavielle, and Eric Moulines. Convergence of a Stochastic Approximation Version of the EM Algorithm. *Annals of Statistics*, pages 94–128, 1999. 2

[16] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 39: 2–38, 1977. 2

[17] Denis Dochain. State and Parameter Estimation in Chemical and Biochemical Processes: A Tutorial. *Journal of Process Control*, 13(8):801–818, December 2003. ISSN 0959-1524. doi: 10.1016/S0959-1524(03)00026-X. 3

[18] Connor Duffin, Edward Cripps, Thomas Stemler, and Mark Girolami. Statistical Finite Elements for Misspecified Models. *Proceedings of the National Academy of Sciences*, 118(2), January 2021. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.2015006118. 1, 2

[19] Alain Durmus and Éric Moulines. Nonasymptotic Convergence Analysis for the Unadjusted Langevin Algorithm. *Annals of Applied Probability*, 27(3):1551–1587, 2017. 2

[20] Alain Durmus and Éric Moulines. High-dimensional Bayesian Inference via the Unadjusted Langevin Algorithm. *Bernoulli. Official Journal of the Bernoulli Society for Mathematical Statistics and Probability*, 25(4A):2854–2882, 2019. 5

[21] Paula Cordero Encinar, Francesca R Crucinio, and O Deniz Akyildiz. Proximal Interacting Particle Langevin Algorithms. *arXiv preprint arXiv:2406.14292*, 2024. 3, 6

[22] Yildirim Serhat Erdogan, Mustafa Gul, F Necati Catbas, and Pelin Gundes Bakir. Investigation of Uncertainty Changes in Model Outputs for Finite-Element Model Updating using Structural Health Monitoring Data. *Journal of Structural Engineering*, 140(11):04014078, 2014. 3

[23] Alexandre Ern and Jean-Luc Guermond. *Theory and Practice of Finite Elements*, volume 159. Springer, 2004. 5

[24] Lawrence C Evans. *Partial Differential Equations*, volume 19. American Mathematical Soc., 2010. 4

[25] Geir Evensen. The Ensemble Kalman Filter for Combined State and Parameter Estimation. *IEEE Control Systems Magazine*, 29(3):83–104, June 2009. ISSN 1941-000X. doi: 10.1109/MCS.2009.932223. 3

[26] Eky Febrianto, Liam Butler, Mark Girolami, and Fehmi Cirak. Digital Twinning of Self-Sensing Structures Using the Statistical Finite Element Method. *Data-Centric Engineering*, 3:e31, January 2022. ISSN 2632-6736. doi: 10.1017/dce.2022.28. 2

[27] Mark Girolami and Ben Calderhead. Riemann Manifold Langevin and Hamiltonian Monte Carlo Methods. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 73(2):123–214, 2011. 5, 6

[28] Mark Girolami, Eky Febrianto, Ge Yin, and Fehmi Cirak. The Statistical Finite Element Method (statFEM) for Coherent Synthesis of Observation Data and Model Predictions. *Computer Methods in Applied Mechanics and Engineering*, 375:113533, March 2021. ISSN 0045-7825. doi: 10.1016/j.cma.2020.113533. 1, 2

[29] Gustaf Hendeby and Fredrik Gustafsson. On Nonlinear Transformations of Gaussian Distributions. *Linköping University, SE-581*, 83, 2007. 15, 25

[30] Thomas JR Hughes. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Courier Corporation, 2003. 5

[31] M. R. Jones, T. J. Rogers, and E. J. Cross. Constraining Gaussian Processes for Physics-Informed Acoustic Emission Mapping. *Mechanical Systems and Signal Processing*, 188:109984, April 2023. ISSN 0888-3270. doi: 10.1016/j.ymssp.2022.109984. 25

[32] Nikolas Kantas, Arnaud Doucet, Sumeetpal S. Singh, Jan Maciejowski, and Nicolas Chopin. On Particle Methods for Parameter Estimation in State-Space Models. *Statistical Science*, 30(3):328–351, August 2015. ISSN 0883-4237, 2168-8745. doi: 10.1214/14-STS511. 3

[33] Toni Karvonen, Fehmi Cirak, and Mark Girolami. Error Analysis for a Statistical Finite Element Method. *arXiv preprint arXiv:2201.07543*, 2022. 2, 9

[34] Juan Kuntz, Jen Ning Lim, and Adam M. Johansen. Particle Algorithms for Maximum Likelihood Training of Latent Variable Models. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, pages 5134–5180. PMLR, April 2023. 2, 3, 6

[35] Jen Ning Lim, Juan Kuntz, Samuel Power, and Adam M Johansen. Momentum Particle Maximum Likelihood. *arXiv preprint arXiv:2312.07335*, 2023. 16

[36] Hamid Moradkhani, Soroosh Sorooshian, Hoshin V. Gupta, and Paul R. Houser. Dual State–Parameter Estimation of Hydrological Models Using Ensemble Kalman Filter. *Advances in Water Resources*, 28(2):135–147, February 2005. ISSN 0309-1708. doi: 10.1016/j.advwatres.2004.09.002. 3

[37] RDBA Nassar, DBA Jones, SS Kulawik, JR Worden, KW Bowman, Robert Joseph Andres, Parvadha Suntharalingam, JM Chen, CAM Brenninkmeijer, TJ Schuck, et al. Inverse Modeling of CO_2 Sources and Sinks using Satellite Observations of CO_2 from TES and Surface Flask Measurements. *Atmospheric Chemistry and Physics*, 11(12):6029–6047, 2011. 3

[38] Paul Felix Valsecchi Oliva and O Deniz Akyildiz. Kinetic Interacting Particle Langevin Monte Carlo. *arXiv preprint arXiv:2407.05790*, 2024. 3, 6, 16, 22

[39] Yanni Papandreou, Jon Cockayne, Mark Girolami, and Andrew Duncan. Theoretical Guarantees for the Statistical Finite Element Method. *SIAM/ASA Journal on Uncertainty Quantification*, 11(4):1278–1307, December 2023. doi: 10.1137/21M1463963. 2, 9

[40] Eckhard Platen and Nicola Bruti-Liberati. *Numerical Solution of Stochastic Differential Equations with Jumps in Finance*, volume 64. Springer Science & Business Media, 2010. 5

[41] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, Mass., 3. print edition, 2008. ISBN 978-0-262-18253-9. 26

[42] Daniel Sanz-Alonso. *Inverse Problems and Data Assimilation*. Cambridge University Press, Cambridge New York, NY Port Melbourne New Delhi Singapore, 1st edition edition, August 2023. ISBN 978-1-00-941429-6. 1

[43] Louis Sharrock, Daniel Dodd, and Christopher Nemeth. Tuning-Free Maximum Likelihood Training of Latent Variable Models via Coin Betting. In *International Conference on Artificial Intelligence and Statistics*, pages 1810–1818. PMLR, 2024. 16

[44] Arno Solin and Simo Särkkä. Hilbert Space Methods for Reduced-Rank Gaussian Process Regression. *Statistics and Computing*, 30(2):419–446, March 2020. ISSN 1573-1375. doi: 10.1007/s11222-019-09886-w. 5, 25

[45] A. M. Stuart. Inverse Problems: A Bayesian Perspective. *Acta Numerica*, 19:451–559, May 2010. ISSN 0962-4929, 1474-0508. doi: 10.1017/S0962492910000061. 1

[46] Albert Tarantola. *Inverse Problem Theory and Methods for Model Parameter Estimation*. Society for Industrial and Applied Mathematics, January 2005. ISBN 978-0-89871-572-9 978-0-89871-792-1. doi: 10.1137/1.9780898717921. 1

[47] Eric A. Wan and Alex T. Nelson. Dual Kalman Filtering Methods for Nonlinear Prediction, Smoothing and Estimation. In *Advances in Neural Information Processing Systems*, pages 793–799, 1997. 3

[48] Eric A. Wan and Rudolph Van Der Merwe. The Unscented Kalman Filter for Nonlinear Estimation. In *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No. 00EX373)*, pages 153–158. Ieee, 2000. 3

[49] Greg CG Wei and Martin A Tanner. A Monte Carlo Implementation of the EM Algorithm and the Poor Man's Data Augmentation Algorithms. *Journal of the American Statistical Association*, 85(411): 699–704, 1990. 2

[50] Max Welling and Yee W Teh. Bayesian Learning via Stochastic Gradient Langevin Dynamics. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 681–688. Citeseer, 2011. 6

[51] Ying Zhang, Ömer Deniz Akyildiz, Theodoros Damoulas, and Sotirios Sabanis. Nonasymptotic Estimates for Stochastic Gradient Langevin Dynamics under Local Conditions in Nonconvex Optimization. *Applied Mathematics & Optimization*, 87(2):25, 2023. 16

A Proof of Lemma 1

For a symmetric block matrix M of the form

$$M = \begin{bmatrix} A & B \\ B^\top & C \end{bmatrix}$$

we can use the Schur complement to determine the characteristics of this matrix in terms of its blocks, namely the fact it is positive definite (PD). We use the following from [8][Section A.5.5]:

$$\begin{aligned} &(1) \quad M \succ 0 \iff C \succ 0 \text{ and } A - BC^{-1}B^\top \succ 0, \\ &(2) \quad \text{If } C \succ 0, \text{ then, } M \succeq 0 \iff A - BC^{-1}B^\top \succeq 0. \end{aligned}$$

Now for the Hessian matrix we can assess if it is positive definite or positive semi definite (PSD) using these conditions, which will determine is we have $\mu > 0$ for strong convexity. Define

$$\nabla^2 \Psi_\theta^y = \begin{bmatrix} \mathbf{A}_\theta^\top \mathbf{G}^{-1} \mathbf{A}_\theta + \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H} & -\mathbf{A}_\theta^\top \mathbf{G}^{-1} \\ -\mathbf{G}^{-1} \mathbf{A}_\theta & \mathbf{G}^{-1} + \mathbf{\Sigma}^{-1} \end{bmatrix}$$

Now since $\mathbf{G}, \mathbf{\Sigma}$ are covariance matrices, they are symmetric positive definite (SPD), i.e. $\mathbf{G} \succ 0, \mathbf{\Sigma} \succ 0$, as are their inverses $\mathbf{G}^{-1} \succ 0, \mathbf{\Sigma}^{-1} \succ 0$, which implies $\mathbf{G}^{-1} + \mathbf{\Sigma}^{-1} \succ 0$. This meets the condition of $C \succ 0$ in (1), (2). Now we can check if the Schur complement is PD or PSD which will imply PD or PSD for the Hessian $\nabla^2 \Psi_\theta^y$. Denote by $\mathbf{S}_\Psi$ the Schur complement of $\nabla^2 \Psi_\theta^y$:

$$\begin{aligned} \mathbf{S}_\Psi &= \mathbf{A}_\theta^\top \mathbf{G}^{-1} \mathbf{A}_\theta + \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H} - \mathbf{A}_\theta^\top \mathbf{G}^{-1} [\mathbf{G}^{-1} + \mathbf{\Sigma}^{-1}]^{-1} \mathbf{G}^{-1} \mathbf{A}_\theta \\ &= \mathbf{A}_\theta^\top \left[\mathbf{G}^{-1} - \mathbf{G}^{-1} [\mathbf{G}^{-1} + \mathbf{\Sigma}^{-1}]^{-1} \mathbf{G}^{-1} \right] \mathbf{A}_\theta + \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H} \\ &= \mathbf{A}_\theta^\top \left[\mathbf{G} + \mathbf{\Sigma} \right]^{-1} \mathbf{A}_\theta + \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H}, \end{aligned}$$

where we have applied Woodbury's matrix inversion lemma

$$\mathbf{G}^{-1} - \mathbf{G}^{-1} [\mathbf{G}^{-1} + \mathbf{\Sigma}^{-1}]^{-1} \mathbf{G}^{-1} = \left[\mathbf{G} + \mathbf{\Sigma} \right]^{-1},$$

for the second equality. Now we show that $\mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H} \succeq 0$ since $\mathbf{R}^{-1}$ is SPD, which means it admits a Cholesky decomposition, $\mathbf{R}^{-1} = \mathbf{L}_\mathbf{R} \mathbf{L}_\mathbf{R}^\top$, giving

$$\mathbf{x}^\top \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H} \mathbf{x} = \mathbf{x}^\top \mathbf{H}^\top \mathbf{L}_\mathbf{R} \mathbf{L}_\mathbf{R}^\top \mathbf{H} \mathbf{x} = \| \mathbf{L}_\mathbf{R}^\top \mathbf{H} \mathbf{x} \|_2^2 \geq 0, \quad \forall \mathbf{x} \in \mathbb{R}^{n_u}, \mathbf{x} \neq 0.$$

We can also show that $\mathbf{A}_\theta^\top [\mathbf{G} + \mathbf{\Sigma}]^{-1} \mathbf{A}_\theta$ is PD, as $[\mathbf{G} + \mathbf{\Sigma}]^{-1}$ is SPD and $\mathbf{A}_\theta$ is non-singular. We define $\mathbf{x} = \mathbf{A}_\theta \mathbf{y}$

$$\mathbf{x}^\top \mathbf{A}_\theta^\top \left[\mathbf{G} + \mathbf{\Sigma} \right]^{-1} \mathbf{A}_\theta \mathbf{x} = \mathbf{y}^\top \mathbf{L} \mathbf{L}^\top \mathbf{y} = \| \mathbf{L}^\top \mathbf{y} \|_2^2 > 0, \quad \forall \mathbf{y} \in \mathbb{R}^{n_u}, \mathbf{y} \neq 0,$$

which holds because we have for every $\mathbf{x} \in \mathbb{R}^{n_u}$, we have $\mathbf{y} = \mathbf{A}_\theta^{-1} \mathbf{x}$. Putting this together, we can show the Schur complement of the Hessian is SPD:

$$\begin{aligned} \mathbf{x}^\top \mathbf{S}_\Psi \mathbf{x} &= \mathbf{x}^\top \mathbf{A}_\theta^\top \left[\mathbf{G} + \mathbf{\Sigma} \right]^{-1} \mathbf{A}_\theta \mathbf{x} + \mathbf{x}^\top \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H} \mathbf{x} \\ &\geq \mathbf{x}^\top \mathbf{A}_\theta^\top \left[\mathbf{G} + \mathbf{\Sigma} \right]^{-1} \mathbf{A}_\theta \mathbf{x} \\ &> 0, \end{aligned}$$

which implies $\nabla^2 \Psi_\theta^y \succ 0$ from (1), hence $\exists \mu > 0$ where $\mu = \lambda_{\min}(\nabla^2 \Psi_\theta^y)$. It is also worth noting that without the inclusion of the prior on $\mathbf{b}$, when just considering the MMLE potential Φ_θ^y, we have

$$\nabla^2 \Phi_\theta^y = \begin{bmatrix} \mathbf{A}_\theta^\top \mathbf{G}^{-1} \mathbf{A}_\theta + \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H} & -\mathbf{A}_\theta^\top \mathbf{G}^{-1} \\ -\mathbf{G}^{-1} \mathbf{A}_\theta & \mathbf{G}^{-1} \end{bmatrix},$$

and the Schur complement becomes

$$\begin{aligned}\mathbf{S}_\Phi &= \mathbf{A}_\theta^\top \mathbf{G}^{-1}\mathbf{A}_\theta + \mathbf{H}^\top\mathbf{R}^{-1}\mathbf{H} - \mathbf{A}_\theta^\top\mathbf{G}^{-1}\mathbf{G}\mathbf{G}^{-1}\mathbf{A}_\theta \\ &= \mathbf{H}^\top\mathbf{R}^{-1}\mathbf{H} \succeq 0,\end{aligned}$$

which we have shown is only PSD, which implies via (2) that $\nabla^2\Phi_\theta^y \succeq 0$, and hence there may not exist a positive minimum eigenvalue of the Hessian, which only implies convexity, and not *strong* convexity.

B Proof of Theorem 1

Our proof is based on Theorem 1 of [13]. It involves re-writing IPLA as an instance of ULA and then invoking standard theoretical guarantees. Let us define a target measure on $\mathbb{R}^{(N+1)n_u}$

$$\tilde{\pi}(\mathbf{b}, \mathbf{p}_1, \ldots, \mathbf{p}_N) \propto \exp\left(-\sum_{i=1}^N \Psi_\theta^y(\sqrt{N}\mathbf{p}_i, \mathbf{b})\right). \tag{49}$$

It is clear that $\mathbf{b}$-marginal of this measure will concentrate on the MMAP solution as $\tilde{\pi}_{\mathbf{b}}(\mathbf{b}) \propto \exp(N \log p(\mathbf{b}|\mathbf{y}))$ by computing $\tilde{\pi}_{\mathbf{b}}(\mathbf{b}) = \int \tilde{\pi}(\mathbf{b}, \mathbf{p}_1, \ldots, \mathbf{p}_N)\mathrm{d}\mathbf{p}_1 \ldots \mathrm{d}\mathbf{p}_N$. It is also clear that the potential $\Psi_{\theta,N}^y(\mathbf{b}, \mathbf{p}_1, \ldots, \mathbf{p}_N) = \sum_{i=1}^N \Psi_\theta^y(\sqrt{N}\mathbf{p}_i, \mathbf{b})$ is $\tilde{\mu} = N\mu$ strongly convex and $\tilde{L} = NL$ gradient Lipschitz based on Lemma 1 and [38][Lemma A.4 and A.5]. The factor of N comes as a result of the potential being a sum of N potentials with Lipschitz and strong convexity constants L, μ, and hence $N\mu \preceq \nabla^2\Psi_{\theta,N}^y \preceq NL$. Now, let us write the standard ULA for the target (49)

$$\begin{aligned}\mathrm{d}\mathbf{b}_t &= -\sum_{n=1}^N \nabla_{\mathbf{b}}\Psi_\theta^y(\sqrt{N}\mathbf{p}_t^{(n)}, \mathbf{b}_t)\mathrm{d}t + \sqrt{2}\mathrm{d}\mathbf{B}_t^0, \\ \mathrm{d}\mathbf{p}_t^{(n)} &= -\sqrt{N}\nabla_1\Psi_\theta^y(\sqrt{N}\mathbf{p}_t^{(n)}, \mathbf{b}_t)\mathrm{d}t + \sqrt{2}\mathrm{d}\mathbf{B}_t^n, \quad n = 1, \ldots, N,\end{aligned}$$

where ∇_1 denotes the gradient w.r.t. the first argument of Ψ_θ^y and $(\mathbf{B}_t^n)_{t\geq 0}$ for $n = 0, \ldots, N$ are n_u-dimensional Brownian motions. Consider the Euler-Maruyama discretisation of this system with a step-size $\tilde{\gamma}$

$$\begin{aligned}\mathbf{b}_{k+1} &= \mathbf{b}_k - \tilde{\gamma}\sum_{n=1}^N \nabla_{\mathbf{b}}\Psi_\theta^y(\sqrt{N}\mathbf{p}_k^{(n)}, \mathbf{b}_k) + \sqrt{2\tilde{\gamma}}\mathbf{w}_{k+1}^0, \\ \mathbf{p}_{k+1}^{(n)} &= \mathbf{p}_k^{(n)} - \tilde{\gamma}\sqrt{N}\nabla_1\Psi_\theta^y(\sqrt{N}\mathbf{p}_k^{(n)}, \mathbf{b}_k) + \sqrt{2\tilde{\gamma}}\mathbf{w}_{k+1}^n, \quad n = 1, \ldots, N,\end{aligned}$$

where $\mathbf{w}_k^n \sim \mathcal{N}(0, I_{n_u})$ for every $k \geq 0$ and $n = 0, \ldots, N$. It is clear that, if we write $\mathbf{u}_k^{(n)} = \sqrt{N}\mathbf{p}_k^{(n)}$, and set $\tilde{\gamma} = \gamma/N$ then the above system is equivalent to IPLA given in eqs. (24)–(25). We now apply Theorem 1 of [13] directly with the condition $\tilde{\gamma} \leq 2/(\tilde{\mu} + \tilde{L})$, results in the bound

$$W_2(\nu_k, \tilde{\pi}) \leq (1 - \tilde{\mu}\tilde{\gamma})^k W_2(\nu_0, \tilde{\pi}) + 1.65\frac{\tilde{L}}{\tilde{\mu}}\sqrt{(N+1)n_u}\tilde{\gamma}^{1/2}.$$

At this point, one must note the restriction can be rewritten as follows

$$\tilde{\gamma} \leq 2/(\tilde{\mu} + \tilde{L}) \iff \gamma \leq 2/(\mu + L).$$

Hence in practice it suffices to assume $\gamma \leq 2/(\mu + L)$. Similarly, the bound above can be rewritten by substituting $\tilde{L} = NL$, $\tilde{\mu} = N\mu$, and $\tilde{\gamma} = \gamma/N$,

$$W_2(\nu_k, \tilde{\pi}) \leq (1 - \mu\gamma)^k W_2(\nu_0, \tilde{\pi}) + 1.65\frac{L}{\mu}\sqrt{\frac{(N+1)n_u}{N}}\gamma^{1/2}.$$

Merging this with Proposition 3 in [3] which ensures that

$$W_2(\delta_{\mathbf{b}^\star}, \tilde{\pi}_{\mathbf{b}}) \leq \sqrt{\frac{2n_u}{\mu N}},$$

and the fact that $W_2(\nu_{k,\mathbf{b}}, \tilde{\pi}_{\mathbf{b}}) \leq W_2(\nu_k, \tilde{\pi})$, we obtain our bound by using

$$\begin{aligned}\mathbb{E}[\|\mathbf{b}_k - \mathbf{b}^\star\|^2]^{1/2} = W_2(\nu_{k,\mathbf{b}}, \delta_{\mathbf{b}^\star}) &\leq W_2(\nu_{k,\mathbf{b}}, \tilde{\pi}_{\mathbf{b}}) + W_2(\tilde{\pi}_{\mathbf{b}}, \delta_{\mathbf{b}^\star}) \\ &\leq W_2(\nu_k, \tilde{\pi}) + W_2(\tilde{\pi}_{\mathbf{b}}, \delta_{\mathbf{b}^\star}),\end{aligned}$$

from which the bound follows.

C Forcing estimation for Linear PDE - analytic solution

To derive the MMAP forcing estimate, we start by marginalising out the latent variables $\mathbf{u}$ from the joint distribution $p(\mathbf{y}|\theta, \mathbf{b}) = \int p(\mathbf{y}|\mathbf{u})p(\mathbf{u}|\theta, \mathbf{b})\mathrm{d}\mathbf{u}$. With models $\mathbf{A}_\theta \mathbf{u} = \mathbf{b} + \mathbf{e}$ and $\mathbf{y} = \mathbf{H}\mathbf{u} + \mathbf{r}$, where $\mathbf{e} \sim \mathcal{N}(\mathbf{0}, \mathbf{G})$ and $\mathbf{r} \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$ we can marginalise analytically giving $p(\mathbf{y}|\theta, \mathbf{b}) = \mathcal{N}\left(\mathbf{H}\mathbf{A}_\theta^{-1}\mathbf{b}, \mathbf{H}\mathbf{A}_\theta^{-1}\mathbf{G}\mathbf{A}_\theta^{-\top}\mathbf{H}^\top + \mathbf{R}\right)$. With the conjugate prior $p(\mathbf{b}) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, we can calculate the posterior $p(\mathbf{b}|\mathbf{y}, \theta) \propto p(\mathbf{y}|\theta, \mathbf{b})p(\mathbf{b})$, where

$$\begin{aligned}\log p(\mathbf{b}|\mathbf{y}) = -\frac{1}{2}\left(\mathbf{y} - \mathbf{H}\mathbf{A}_\theta^{-1}\mathbf{b}\right)^\top \left[\mathbf{H}\mathbf{A}_\theta^{-1}\mathbf{G}\mathbf{A}_\theta^{-\top}\mathbf{H}^\top + \mathbf{R}\right]^{-1}\left(\mathbf{y} - \mathbf{H}\mathbf{A}_\theta^{-1}\mathbf{b}\right) \\ -\frac{1}{2}\left(\mathbf{b} - \boldsymbol{\mu}\right)^\top \boldsymbol{\Sigma}^{-1}\left(\mathbf{b} - \boldsymbol{\mu}\right) + \text{const},\end{aligned} \tag{50}$$

where MMAP estimator, $\mathbf{b}^\star = \text{argmax}_{\mathbf{b}}\, p(\mathbf{b}|\theta, \mathbf{y})$ follows as the mean of the Gaussian posterior $p(\mathbf{b}|\theta, \mathbf{y})$:

$$\begin{aligned}\mathbf{b}^\star = \left[\mathbf{A}_\theta^{-\top}\mathbf{H}^\top\left[\mathbf{H}\mathbf{A}_\theta^{-1}\mathbf{G}\mathbf{A}_\theta^{-\top}\mathbf{H}^\top + \mathbf{R}\right]^{-1}\mathbf{H}\mathbf{A}_\theta^{-1} + \boldsymbol{\Sigma}^{-1}\right]^{-1} \\ \left[\mathbf{A}_\theta^{-\top}\mathbf{H}^\top\left[\mathbf{H}\mathbf{A}_\theta^{-1}\mathbf{G}\mathbf{A}_\theta^{-\top}\mathbf{H}^\top + \mathbf{R}\right]^{-1}\mathbf{y} + \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}\right].\end{aligned} \tag{51}$$

For calculating the true posterior variance in Figure 3, with $p(\mathbf{u}|\mathbf{y}, \mathbf{b}^\star) \sim \mathcal{N}(\mathbf{m}_{\mathbf{u}|\mathbf{y}}, \mathbf{C}_{\mathbf{u}|\mathbf{y}})$:

$$\mathbf{C}_{\mathbf{u}|\mathbf{y}} = \left[\mathbf{A}_\theta^\top \mathbf{G}^{-1}\mathbf{A}_\theta + \mathbf{H}^\top\mathbf{R}^{-1}\mathbf{H}\right]^{-1}, \quad \mathbf{m}_{\mathbf{u}|\mathbf{y}} = \mathbf{C}_{\mathbf{u}|\mathbf{y}}(\mathbf{H}^\top\mathbf{R}^{-1}\mathbf{y} + \mathbf{A}_\theta^\top\mathbf{G}^{-1}\mathbf{b}^\star). \tag{52}$$

D Nonlinear Poisson discretisation

The strong form of the nonlinear Poisson equation with Dirichlet boundary conditions reads as

$$\begin{cases} -\nabla\cdot(q(u)\nabla u) = f + \epsilon & \text{in } \Omega \\ u = u_D & \text{on } \partial\Omega. \end{cases}$$

We wish to find solutions in the Sobolev space, respecting the Dirichlet boundary conditions, $V = \left\{v \in H_0^1(\Omega)\colon v = u_D \text{ on } \partial\Omega\right\}$. Converting to the weak form proceeds as before, by multiplication with a test function $v \in \hat{V}$, where $\hat{V} = \left\{v \in H_0^1(\Omega)\colon v = 0 \text{ on } \partial\Omega\right\}$ and integrating over the domain

$$-\int_\Omega \nabla\cdot(q(u)\nabla u)v\mathrm{d}x = \int_\Omega (f+\epsilon)\, v\mathrm{d}x. \tag{53}$$

The product rule for divergence can be used to rewrite the left hand side integrand as follows

$$\nabla\cdot(q(u)\nabla u)v = \nabla\cdot(q(u)\nabla uv) - q(u)\nabla u\cdot\nabla v. \tag{54}$$

Substituting (54) into (53) gives

$$-\int_\Omega \nabla\cdot(q(u)\nabla u v)\mathrm{d}x + \int_\Omega q(u)\nabla u\cdot\nabla v\mathrm{d}x = \int_\Omega (f+\epsilon)\,v\mathrm{d}x. \tag{55}$$

The first term in (55) can be expressed as a surface integral via the divergence theorem [9], and since $v=0$ on $\partial\Omega$, this term vanishes

$$\int_\Omega \nabla\cdot(q(u)\nabla u v)\mathrm{d}x = \int_{\partial\Omega}(q(u)\nabla u v)\cdot\nu ds = 0, \tag{56}$$

where ν denotes the outward normal vector to $\partial\Omega$. The nonlinear weak form

$$\int_\Omega q(u)\nabla u\cdot\nabla v\mathrm{d}x = \int_\Omega (f+\epsilon)\,v\mathrm{d}x. \tag{57}$$

Again we specify a basis function expansion of the solution $u_h = \sum_{j=1}^{n_u}\hat{u}_j\phi_j$, and search for solutions in the space $V_h = \text{span}\,\{\phi_j\}_{j=1}^{n_u}$, and test with the basis functions individually giving a system of nonlinear algebraic equations in the FEM coefficients,

$$\int_\Omega q(u_h)\nabla u_h\cdot\nabla\phi_j\mathrm{d}x = \int_\Omega (f+\epsilon)\,\phi_j\mathrm{d}x, \quad \forall j\in\{1,\ldots,n_u\}, \tag{58}$$

which can be written in vector form

$$\mathbf{F}(\mathbf{u}) = \mathbf{b}+\mathbf{e}, \quad \mathbf{e}\sim\mathcal{N}(\mathbf{0},\mathbf{G}), \tag{59}$$

where we have $\mathbf{u} = [\hat{u}_1,\hat{u}_2,\ldots,\hat{u}_{n_u}]^\top$, $[\mathbf{F}(\mathbf{u})]_i = \langle q(u_h)\nabla u_h,\nabla\phi_i\rangle$, $[\mathbf{b}]_i = \langle f,\phi_i\rangle$, and $[\mathbf{G}]_{ij} = \langle\phi_i,\langle k(\cdot,\cdot),\phi_j\rangle\rangle$. Section 6.2 describes the approximations made to the statFEM prior, all of which require the system (59) to be solved. This discrete problem is efficiently solved using Newton's method.

E Approximations to Nonlinearly Transformed Gaussian Random Variables

Here we present the three methods for deriving a Gaussian approximation to a nonlinearly transformed Gaussian random variable. The transform we consider is of the form $\mathbf{F}(\mathbf{u}) = \mathbf{b}+\mathbf{e}$, where we wish to approximate $\mathbf{u}$ with a Gaussian, given $\mathbf{e}\sim\mathcal{N}(\mathbf{0},\mathbf{G})$.

E.1 First-order Taylor

We linearise the system about the mean point $\mathbf{m}$, found as the solution of $\mathbf{F}(\mathbf{m}) = \mathbf{b}$. By replacing $\mathbf{u}$ in (59) with $\mathbf{m}$, as follows

$$\mathbf{F}(\mathbf{u})-\mathbf{b}\approx\mathbf{J}(\mathbf{m})(\mathbf{u}-\mathbf{m}), \quad \text{giving} \quad \mathbf{u} = \mathbf{m}+\mathbf{J}(\mathbf{m})^{-1}\mathbf{e}, \quad \mathbf{e}\sim\mathcal{N}(\mathbf{0},\mathbf{G}). \tag{60}$$

Computing the second moment,

$$\mathbf{C} = \mathbb{E}\left[(\mathbf{u}-\mathbf{m})(\mathbf{u}-\mathbf{m})^\top\right] = \mathbf{J}(\mathbf{m})^{-1}\mathbb{E}\left[\mathbf{e}\mathbf{e}^\top\right]\mathbf{J}(\mathbf{m})^{-\top} = \mathbf{J}(\mathbf{m})^{-1}\mathbf{G}\mathbf{J}(\mathbf{m})^{-\top}, \tag{61}$$

we arrive at the approximate Gaussian distribution $\mathbf{u}\sim\mathcal{N}(\mathbf{m},\mathbf{C})$.

E.2 Unscented transform

Selecting the sigma-points to evaluate the nonlinear transform requires the singular value decomposition of the covariance $\mathbf{G} = \sum_{j=1}^{n_u} \sigma_j^2 \mathbf{v}_j \mathbf{v}_j^\top$, and parameters α, β

$$\mathbf{s}^{(0)} = \mathbf{b}, \quad \mathbf{s}^{(\pm j)} = \mathbf{b} \pm \sqrt{n_u + \lambda}\sigma_j \mathbf{v}_j \tag{62}$$

$$w^{(0)} = \frac{\lambda}{n_u + \lambda}, \quad w^{(\pm j)} = \frac{1}{2(n_u + \lambda)} \tag{63}$$

where $j = 1, \ldots, n_u$ and $\lambda = \alpha^2(n_u + \kappa) - n_u$. The parameter α controls the spread of the sigma points and is typically chosen as 10^{-3}, and β chosen as $\beta = 2$ (See [29]).

$$\mathbf{m} = \sum_{j=-n_u}^{n_u} w^{(j)} \mathbf{u}^{(j)}, \quad \mathbf{C} = \sum_{j=-n_u}^{n_u} \hat{w}^{(j)} (\mathbf{u}^{(j)} - \mathbf{m})(\mathbf{u}^{(j)} - \mathbf{m})^\top, \tag{64}$$

where $\hat{w}^{(0)} = w^{(0)} + 1 - \alpha^2 + \beta$ and $\hat{w}^{(j)} = w^{(j)}$ for $j \neq 0$.

E.3 Monte Carlo

The first two moments of the non-Gaussian statFEM prior are approximated using Monte Carlo samples. The approximation is of the form $p(\mathbf{u}|\mathbf{b}) \approx \mathcal{N}(\mathbf{m}, \mathbf{C})$, samples are generated via M forward solves of the PDE where the noise is sampled from its Gaussian distribution, $\mathbf{F}(\mathbf{u}^{(j)}) = \mathbf{b} + \mathbf{e}^{(j)}$ with, $\mathbf{e}^{(j)} \sim \mathcal{N}(\mathbf{0}, \mathbf{G}), \forall j \in \{1, \ldots, M\}$ giving the Monte Carlo (MC) samples $\left\{\mathbf{u}^{(j)}\right\}_{j=1}^{M}$. The statistics can then be computed empirically

$$\mathbf{m} = \frac{1}{M} \sum_{j=1}^{M} \mathbf{u}^{(j)}, \quad \mathbf{C} = \frac{1}{M-1} \sum_{j=1}^{M} (\mathbf{u}^{(j)} - \mathbf{m})(\mathbf{u}^{(j)} - \mathbf{m})^\top. \tag{65}$$

F Hilbert space GP

In this section we show how the Hilbert space GP covariance function is approximated with an FEM basis. To determine the functions $g_l(x)$ that form the covariance approximation in (11), we must solve the eigenfunction problem for the Laplacian [31, 44], with homogeneous Dirichlet boundary conditions

$$\forall l \in \{1, \ldots, n\}, \quad \text{solve} \begin{cases} -\nabla^2 g_l(x) = \lambda_l g_l(x), & x \in \Omega \\ g_l(x) = 0, & x \in \partial\Omega. \end{cases} \tag{66}$$

We wish to solve for eigenpairs $\{\lambda_l, g_l\}_{l=1}^{n}$. For a generic mesh defined over a domain Ω, we can solve this problem using FEM. With our FEM basis defined as $\{\phi_i\}_{i=1}^{n}$, we can take the inner product with the j^{th} basis function, and application of integration by parts gives us the weak form

$$\int_\Omega \nabla g_l(x) \cdot \nabla \phi_j(x) \mathrm{d}x = \int_\Omega \lambda_l g_l(x) \phi_j(x) \mathrm{d}x. \tag{67}$$

If we assume a basis function approximation of the eigenfunction, i.e. $\hat{g}_l(x) = \sum_{i=1}^{n} \hat{g}_{li} \phi_i(x)$, we can write this as

$$\sum_{i=1}^{n} \hat{g}_{li} \int_\Omega \nabla \phi_i(x) \cdot \nabla \phi_j(x) \mathrm{d}x = \lambda_l \sum_{i=1}^{n} \hat{g}_{li} \int_\Omega \phi_i(x) \phi_j(x) \mathrm{d}x, \tag{68}$$

which can be written in the form of a generalised eigenproblem

$$\mathbf{A}\hat{\mathbf{g}}_l = \lambda_l \mathbf{M}\hat{\mathbf{g}}_l, \tag{69}$$

where $[\mathbf{A}]_{ij} = \int_\Omega \nabla\phi_i \cdot \nabla\phi_j \mathrm{d}x$ is the stiffness matrix, $[\mathbf{M}]_{ij} = \int_\Omega \phi_i\phi_j \mathrm{d}x$ the mass matrix, and $\hat{\mathbf{g}}_l = [\hat{g}_{l1}, \ldots, \hat{g}_{ln}]^\top$ are the eigenvectors to be solved for. Equation (69) can be solved using standard linear algebra libraries; here we use the function `scipy.linalg.eig`$(\mathbf{A}, \mathbf{M})$ which provides unit l_2-norm eigenvectors $\hat{\mathbf{g}}_l$. The eigenfunction approximations are converted to unit L^2-norm functions by normalising the coefficients, i.e. $g_l(x) = \sum_{i=1}^n \tilde{g}_{li}\phi_i(x)$ where, $\tilde{g}_{li} = \hat{g}_{li}/\sqrt{\langle \hat{g}_l, \hat{g}_l \rangle}$. In this paper we exclusively use the squared exponential kernel, parameterised by an amplitude σ^2 and length-scale ℓ, which has the following form and associated spectral density [41]

$$k_{\mathrm{SE}}(r) = \sigma^2 \exp\left(-\frac{r^2}{2\ell^2}\right), \quad S_{\mathrm{SE}}(\omega) = \sigma^2(2\pi\ell^2)^{D/2} \exp\left(-\frac{\omega^2\ell^2}{2}\right), \tag{70}$$

where $r = \|x - x'\|$, $x \in \mathbb{R}^D$. Figure 6 visualises the eigenfunctions computed via (69) for the disc geometry.

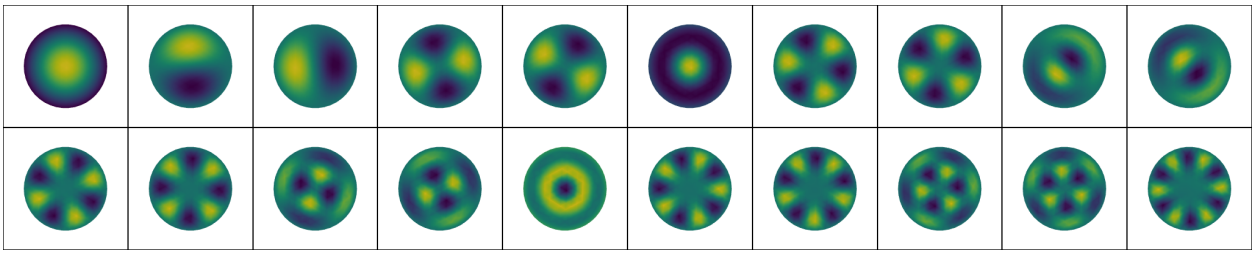

Figure 6: FEM approximations to the eigenfunctions of the Laplacian for the domain $\Omega = \{\mathbf{x} \in \mathbb{R}^2 \colon \|\mathbf{x}\| < 1\}$, for the smallest 20 eigenvalues.