



Solving and learning nonlinear PDEs with Gaussian processes

Yifan Chen, Bamdad Hosseini, Houman Owghadi*, Andrew M. Stuart

Computing and Mathematical Sciences, Caltech, Pasadena, CA, United States of America

ARTICLE INFO

Article history:

Available online 31 August 2021

Keywords:

Kernel methods

Gaussian processes

Nonlinear partial differential equations

Inverse problems

Optimal recovery

ABSTRACT

We introduce a simple, rigorous, and unified framework for solving nonlinear partial differential equations (PDEs), and for solving inverse problems (IPs) involving the identification of parameters in PDEs, using the framework of Gaussian processes. The proposed approach: (1) provides a natural generalization of collocation kernel methods to nonlinear PDEs and IPs; (2) has guaranteed convergence for a very general class of PDEs, and comes equipped with a path to compute error bounds for specific PDE approximations; (3) inherits the state-of-the-art computational complexity of linear solvers for dense kernel matrices. The main idea of our method is to approximate the solution of a given PDE as the maximum a posteriori (MAP) estimator of a Gaussian process conditioned on solving the PDE at a finite number of collocation points. Although this optimization problem is infinite-dimensional, it can be reduced to a finite-dimensional one by introducing additional variables corresponding to the values of the derivatives of the solution at collocation points; this generalizes the representer theorem arising in Gaussian process regression. The reduced optimization problem has the form of a quadratic objective function subject to nonlinear constraints; it is solved with a variant of the Gauss–Newton method. The resulting algorithm (a) can be interpreted as solving successive linearizations of the nonlinear PDE, and (b) in practice is found to converge in a small number of iterations (2 to 10), for a wide range of PDEs. Most traditional approaches to IPs interleave parameter updates with numerical solution of the PDE; our algorithm solves for both parameter and PDE solution simultaneously. Experiments on nonlinear elliptic PDEs, Burgers' equation, a regularized Eikonal equation, and an IP for permeability identification in Darcy flow illustrate the efficacy and scope of our framework.

© 2021 Elsevier Inc. All rights reserved.

1. Introduction

Two hundred years ago, modeling a physical problem and solving the underlying differential equations would have required the expertise of Cauchy or Laplace, and it would have been done by hand through a laborious process of discovery. Sixty years ago, the resolution of the underlying differential equation would have been addressed using computers, but modeling and design of the solver would have still been done by hand. Nowadays, there is increasing interest in automating these steps by casting them as machine learning problems. The resulting methods can be divided into two main categories: (1) methods based on variants of artificial neural networks (ANNs) [22]; and (2) methods based on kernels and Gaussian Processes (GPs) [89,68]. In the context of (1) there has been recent activity toward solving nonlinear PDEs, whilst the systematic development of methods of type (2) for nonlinear PDEs has remained largely open. However, methods of type

* Corresponding author.

E-mail addresses: yifanc@caltech.edu (Y. Chen), bamdadh@caltech.edu (B. Hosseini), owghadi@caltech.edu (H. Owghadi), astuart@caltech.edu (A.M. Stuart).

(2) hold potential for considerable advantages over methods of type (1), both in terms of theoretical analysis and numerical implementation. In this paper, our goal is to develop a simple kernel/GP framework for solving nonlinear PDEs and related inverse problems (IPs); in particular the methodology we introduce has the following properties:

- the proposed collocation setting for solving nonlinear PDEs and IPs is a direct generalization of optimal recovery kernel methods for linear PDEs [43,44,47], and a natural generalization of radial basis function collocation methods [91,88], and of meshless kernel methods [64];
- theoretically, the proposed method is provably convergent and amenable to rigorous numerical analysis, suggesting new research directions to generalize the analysis of linear regression methods [88] to the proposed collocation setting for solving nonlinear PDEs;
- computationally, it inherits the complexity of state-of-the-art solvers for dense kernel matrices, suggesting new research to generalize the work of [65], which developed optimal approximate methods for linear regression, to the proposed collocation setting for solving nonlinear PDEs and IPs;
- for IPs the methodology is closely aligned with methodologies prevalent in the PDE-constrained optimization literature [26] and suggests the need for new computational and theoretical analyses generalizing the standard optimization and Bayesian approaches found in [17,20,28].

Since ANN methods can also be interpreted as kernel methods with kernels learned from data [27,45,90], our framework could also be used for theoretical analysis of such methods.

In Subsection 1.1 we summarize the theoretical foundations and numerical implementation of our method in the context of a simple nonlinear elliptic PDE. In Subsection 1.2 we give a literature review, placing the proposed methodology in the context of other research at the intersection of machine learning and PDEs. The outline of the article is given in Subsection 1.3.

1.1. Summary of the proposed method

For demonstration purposes, we summarize the key ideas of our method for solving a nonlinear second-order elliptic PDE. This PDE will also serve as a running example in Section 3 where we present an abstract framework for general nonlinear PDEs.

Let $d \geq 1$ and Ω be a bounded open domain in \mathbb{R}^d with a Lipschitz boundary $\partial\Omega$. Consider the following nonlinear elliptic equation for u^* :

$$\begin{cases} -\Delta u^*(\mathbf{x}) + \tau(u^*(\mathbf{x})) = f(\mathbf{x}), & \forall \mathbf{x} \in \Omega, \\ u^*(\mathbf{x}) = g(\mathbf{x}), & \forall \mathbf{x} \in \partial\Omega, \end{cases} \quad (1.1)$$

where $f: \Omega \rightarrow \mathbb{R}$, $g: \partial\Omega \rightarrow \mathbb{R}$ and $\tau: \mathbb{R} \rightarrow \mathbb{R}$ are given continuous functions. We assume that f, g, τ are chosen appropriately so that the PDE has a unique classical solution (for abstract theory of nonlinear elliptic PDEs see for example [56,72]). In Subsection 1.1.4 we will present a concrete numerical experiment where $\tau(u) = u^3$ and $g(\mathbf{x}) = 0$.

1.1.1. Optimal recovery

Our proposed method starts with an optimal recovery problem that can also be interpreted as maximum a posterior (MAP) estimation for a GP constrained by a PDE. More precisely, consider a nondegenerate, symmetric, and positive definite kernel function $K: \overline{\Omega} \times \overline{\Omega} \rightarrow \mathbb{R}$ where $\overline{\Omega} := \Omega \cup \partial\Omega$. Let \mathcal{U} be the RKHS associated with K and denote the RKHS norm by $\|\cdot\|$. Let $1 \leq M_\Omega < M < \infty$ and fix M points in $\overline{\Omega}$ such that $\mathbf{x}_1, \dots, \mathbf{x}_{M_\Omega} \in \Omega$ and $\mathbf{x}_{M_\Omega+1}, \dots, \mathbf{x}_M \in \partial\Omega$. Then, our method approximates the solution u^* of (1.1) with a minimizer u^\dagger of the following optimal recovery problem:

$$\begin{cases} \text{minimize } \|u\| \\ \text{s.t. } -\Delta u(\mathbf{x}_m) + \tau(u(\mathbf{x}_m)) = f(\mathbf{x}_m), & \text{for } m = 1, \dots, M_\Omega, \\ u(\mathbf{x}_m) = g(\mathbf{x}_m), & \text{for } m = M_\Omega + 1, \dots, M. \end{cases} \quad (1.2)$$

Here, we assume K is chosen appropriately so that $\mathcal{U} \subset C^2(\Omega) \cap C(\overline{\Omega})$, which ensures the pointwise PDE constraints in (1.2) are well-defined.

A minimizer u^\dagger can be interpreted as a MAP estimator of a GP $\xi \sim \mathcal{N}(0, \mathcal{K})$ ¹ (where \mathcal{K} is the integral operator with kernel K) conditioned on satisfying the PDE at the collocation points $\mathbf{x}_m, 1 \leq m \leq M$. Such a view has been introduced for solving linear PDEs in [43,44] and a closely related approach is studied in [12, Sec. 5.2]; the methodology introduced via (1.2) serves as a prototype for generalization to nonlinear PDEs. Here it is important to note that in the nonlinear case the conditioned GP is no longer Gaussian in general; thus the solution of (1.2) is a MAP estimator only and is not the conditional expectation, except in the case where $\tau(\cdot)$ is a linear function.

¹ This Gaussian prior notation is equivalent to the GP notation $\mathcal{GP}(0, K)$, where K is the covariance function. See further discussions in Subsection 3.4.1.

In the next subsections, we show equivalence of (1.2) and a finite dimensional constrained optimization problem (1.5). This provides existence² of a minimizer to (1.2), as well as the basis for a numerical method to approximate the minimizer, based on solution of an unconstrained finite-dimensional optimization problem (1.6).

1.1.2. Finite-dimensional representation

The key to our numerical algorithm for solving (1.2) is a representer theorem that characterizes u^\dagger via a finite-dimensional representation formula. To achieve this we first reformulate (1.2) as a two level optimization problem:

$$\begin{cases} \text{minimize} & \begin{cases} \text{minimize}_{u \in \mathcal{U}} \|u\| \\ \text{s.t. } u(\mathbf{x}_m) = z_m^{(1)} \text{ and } -\Delta u(\mathbf{x}_m) = z_m^{(2)}, \text{ for } m = 1, \dots, M, \end{cases} \\ \text{s.t. } & \begin{cases} z_m^{(2)} + \tau(z_m^{(1)}) = f(\mathbf{x}_m), & \text{for } m = 1, \dots, M_\Omega, \\ z_m^{(1)} = g(\mathbf{x}_m), & \text{for } m = M_\Omega + 1, \dots, M. \end{cases} \end{cases} \quad (1.3)$$

Denote $\phi_m^{(1)} = \delta_{\mathbf{x}_m}$ and $\phi_m^{(2)} = \delta_{\mathbf{x}_m} \circ (-\Delta)$, where $\delta_{\mathbf{x}}$ is the Dirac delta function centered at \mathbf{x} . We further use the shorthand notation $\boldsymbol{\phi}^{(1)}$ and $\boldsymbol{\phi}^{(2)}$ for the M and M_Ω -dimensional vectors with entries $\phi_m^{(1)}$ and $\phi_m^{(2)}$ respectively, and $\boldsymbol{\phi}$ for the $(M + M_\Omega)$ -dimensional vector obtained by concatenating $\boldsymbol{\phi}^{(1)}$ and $\boldsymbol{\phi}^{(2)}$. Similarly, we write \mathbf{z} for the $(M + M_\Omega)$ -dimensional vector concatenating $\mathbf{z}^{(1)}$ and $\mathbf{z}^{(2)}$.

For a fixed \mathbf{z} , we can solve the first level optimization problem explicitly due to the representer theorem³ (see [47, Sec. 17.8]), which leads to the conclusion that

$$u(\mathbf{x}) = K(\mathbf{x}, \boldsymbol{\phi})K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1}\mathbf{z}; \quad (1.4)$$

here $K(\cdot, \boldsymbol{\phi})$ denotes the $(M + M_\Omega)$ -dimensional vector field with entries $\int K(\cdot, \mathbf{x}')\phi_m(\mathbf{x}')d\mathbf{x}'$ and $K(\boldsymbol{\phi}, \boldsymbol{\phi})$ is the $(M + M_\Omega) \times (M + M_\Omega)$ -matrix with entries $\int K(\mathbf{x}, \mathbf{x}')\phi_m(\mathbf{x})\phi_j(\mathbf{x}')d\mathbf{x}d\mathbf{x}'$ with the ϕ_m denoting the entries of $\boldsymbol{\phi}$. For this solution, $\|u\|^2 = \mathbf{z}^T K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1}\mathbf{z}$, so we can equivalently formulate (1.3) as a finite-dimensional optimization problem:

$$\begin{cases} \text{minimize} & \mathbf{z}^T K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1}\mathbf{z} \\ \text{s.t. } & \begin{cases} z_m^{(2)} + \tau(z_m^{(1)}) = f(\mathbf{x}_m), & \text{for } m = 1, \dots, M_\Omega, \\ z_m^{(1)} = g(\mathbf{x}_m), & \text{for } m = M_\Omega + 1, \dots, M. \end{cases} \end{cases} \quad (1.5)$$

Moreover, using the equation $z_m^{(2)} = f(\mathbf{x}_m) - \tau(z_m^{(1)})$ and the boundary data, we can further eliminate $\mathbf{z}^{(2)}$ and rewrite it as an unconstrained problem:

$$\text{minimize}_{\mathbf{z}_\Omega^{(1)} \in \mathbb{R}^{M_\Omega}} (\mathbf{z}_\Omega^{(1)}, g(\mathbf{x}_{\partial\Omega}), f(\mathbf{x}_\Omega) - \tau(\mathbf{z}_\Omega^{(1)}))K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1} \begin{pmatrix} \mathbf{z}_\Omega^{(1)} \\ g(\mathbf{x}_{\partial\Omega}) \\ f(\mathbf{x}_\Omega) - \tau(\mathbf{z}_\Omega^{(1)}) \end{pmatrix}, \quad (1.6)$$

where we used \mathbf{x}_Ω and $\mathbf{x}_{\partial\Omega}$ to denote the interior and boundary points respectively, $\mathbf{z}_\Omega^{(1)}$ denotes the M_Ω -dimensional vector of the z_i for $i = 1, \dots, M_\Omega$ associated to the interior points \mathbf{x}_Ω while $f(\mathbf{x}_\Omega)$, $g(\mathbf{x}_{\partial\Omega})$ and $\tau(\mathbf{z}_\Omega^{(1)})$ are vectors obtained by applying the corresponding functions to entries of their input vectors. For brevity we have suppressed the transpose signs in the row vector multiplying the matrix from the left in (1.6).

The foregoing considerations lead to the following existence result which underpins⁴ our numerical method for (1.1); furthermore (1.6) provides the basis for our numerical implementations. We summarize these facts:

Theorem 1.1. *The variational problem (1.2) has a minimizer of the form $u^\dagger(\mathbf{x}) = K(\mathbf{x}, \boldsymbol{\phi})K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1}\mathbf{z}^\dagger$, where \mathbf{z}^\dagger is a minimizer of (1.5). Furthermore $u^\dagger(\mathbf{x})$ may be found by solving the unconstrained minimization problem (1.6) for $\mathbf{z}_\Omega^{(1)}$.*

Proof. Problems (1.2), (1.3) and (1.5) are equivalent. It is therefore sufficient to show that (1.5) has a minimizer. Write \mathbf{z}^* for the vector with entries $z_m^{*(1)} = u^*(\mathbf{x}_m)$ and $z_m^{*(2)} = -\Delta u^*(\mathbf{x}_m)$. Since u^* solves the PDE (1.1), \mathbf{z}^* satisfies the constraints

² Uniqueness of a minimizer is not necessarily a consequence of the assumed uniqueness of the solution to the PDE (1.1). We provide additional discussions of uniqueness results in Subsection 5.1.

³ This is not the standard RKHS/GP representer theorem [89, Sec. 2.2] in the sense that measurements include the pointwise observation of higher order derivatives of the underlying GP. See [30] and [84, p. xiii] for related spline representation formulas with derivative information.

⁴ Although this existence result is desirable, it is not necessary in the sense that our variational analysis of minimizers of (1.2), remains true for near-minimizers. Furthermore, the error estimates of Section 5.2 can be modified to include the error incurred by approximating u^* with a near-minimizer rather than a minimizer u^\dagger .

on \mathbf{z} in (1.5). It follows that the minimization in (1.5) can be restricted to the set \mathcal{C} of \mathbf{z} that satisfies $\mathbf{z}^T K(\phi, \phi)^{-1} \mathbf{z} \leq (\mathbf{z}^*)^T K(\phi, \phi)^{-1} \mathbf{z}^*$ and the nonlinear constraints. The set \mathcal{C} is compact and non-empty: compact because τ is continuous and so the constraint set is closed as it is the pre-image of a closed set, and the intersection of a closed set with a compact set is compact; and nonempty because it contains \mathbf{z}^* . Thus the objective function $\mathbf{z}^T K(\phi, \phi)^{-1} \mathbf{z}$ achieves its minimum in the set \mathcal{C} . Once $\mathbf{z}_\Omega^{(1)}$ is found we can extend to the boundary points to obtain $\mathbf{z}^{(1)}$, and use the nonlinear relation between $\mathbf{z}^{(1)}$ and $\mathbf{z}^{(2)}$ to reconstruct $\mathbf{z}^{(2)}$. This gives \mathbf{z}^\dagger . ■

1.1.3. Convergence theory

The consistency of our method is guaranteed by the convergence of u_M^\dagger towards u^* as M , the total number of collocation points, goes to infinity. We first present this result in the case of the nonlinear PDE (1.1) and defer a more general version to Subsection 3.2. We also give a simple proof of convergence here as it is instructive in understanding why the method works and how the more general result can be established. Henceforth we use $|\cdot|$ to denote the standard Euclidean norm and write $\mathcal{U} \subseteq \mathcal{H}$ to denote \mathcal{U} being continuously embedded in Banach space \mathcal{H} .

Theorem 1.2. Suppose that K is chosen so that $\mathcal{U} \subseteq H^s(\Omega)$ for some $s > 2 + d/2$ and that (1.1) has a unique classical solution $u^* \in \mathcal{U}$. Write u_M^\dagger for a minimizer of (1.2) with M distinct collocation points $\mathbf{x}_1, \dots, \mathbf{x}_M$. Assume that, as $M \rightarrow \infty$,

$$\sup_{\mathbf{x} \in \Omega} \min_{1 \leq m \leq M_\Omega} |\mathbf{x} - \mathbf{x}_m| \rightarrow 0 \quad \text{and} \quad \sup_{\mathbf{x} \in \partial\Omega} \min_{M_\Omega+1 \leq m \leq M} |\mathbf{x} - \mathbf{x}_m| \rightarrow 0.$$

Then, as $M \rightarrow \infty$, u_M^\dagger converges towards u^* pointwise in Ω and in $H^t(\Omega)$ for any $t < s$.

Proof. The proof relies on a simple compactness argument together with the Sobolev embedding theorem [1,8]. First, as u^* satisfies the constraint in (1.2) and u_M^\dagger is the minimizer, it follows that $\|u_M^\dagger\| \leq \|u^*\|$ for all $M \geq 1$. This implies $\|u_M^\dagger\|_{H^s(\Omega)} \leq C\|u^*\|$ because \mathcal{U} is continuously embedded into $H^s(\Omega)$. Let $t \in (2 + d/2, s)$ so that $H^t(\Omega)$ embeds continuously into $C^2(\Omega) \cap C(\overline{\Omega})$ [1, Thm. 4.12]. Since $H^s(\Omega)$ is compactly embedded into $H^t(\Omega)$, we deduce that there exists a subsequence $\{M_p, p \geq 1\} \subset \mathbb{N}$ and a limit $u_\infty^\dagger \in H^t(\Omega)$ such that $u_{M_p}^\dagger$ converges towards u_∞^\dagger in the $H^t(\Omega)$ norm, as $p \rightarrow \infty$. This also implies convergence in $C^2(\Omega)$ due to the continuous embedding of $H^t(\Omega)$ into $C^2(\Omega) \cap C(\overline{\Omega})$.

We now show that u_∞^\dagger satisfies the desired PDE in (1.1). Denote by $v = -\Delta u_\infty^\dagger + \tau(u_\infty^\dagger) - f \in C(\Omega)$ and $v_p = -\Delta u_{M_p}^\dagger + \tau(u_{M_p}^\dagger) - f \in C(\Omega)$. For any $\mathbf{x} \in \Omega$ and $p \geq 1$, use of the triangle inequality shows that

$$\begin{aligned} |v(\mathbf{x})| &\leq \min_{1 \leq m \leq M_{p,\Omega}} (|v(\mathbf{x}) - v(\mathbf{x}_m)| + |v(\mathbf{x}_m) - v_p(\mathbf{x}_m)|) \\ &\leq \min_{1 \leq m \leq M_{p,\Omega}} |v(\mathbf{x}) - v(\mathbf{x}_m)| + \|v - v_p\|_{C(\Omega)}, \end{aligned} \quad (1.7)$$

where we have used the fact that $v_p(\mathbf{x}_m) = 0$, and $M_{p,\Omega}$ is the number of interior points associated with the total M_p collocation points. Since v is continuous in a compact domain, it is also uniformly continuous. Thus, it holds that $\lim_{p \rightarrow \infty} \min_{1 \leq m \leq M_{p,\Omega}} |v(\mathbf{x}) - v(\mathbf{x}_m)| = 0$ since the fill-distance converges to zero by assumption. Moreover, we have that v_p converges to v in the $C(\Omega)$ norm due to the $C^2(\Omega)$ convergence from $u_{M_p}^\dagger$ to u_∞^\dagger . Combining these facts with (1.7), and taking $p \rightarrow \infty$, we obtain $v(\mathbf{x}) = 0$, and thus $-\Delta u_\infty^\dagger(\mathbf{x}) + \tau(u_\infty^\dagger(\mathbf{x})) = f(\mathbf{x})$, for $\mathbf{x} \in \Omega$. Following a similar argument, we get $u_\infty^\dagger(\mathbf{x}) = g(\mathbf{x})$ for $\mathbf{x} \in \partial\Omega$. Thus, u_∞^\dagger is a classical solution to (1.1). By assumption, the solution is unique, so we must have $u_\infty^\dagger = u^*$. As the limit u_∞^\dagger is independent of the particular subsequence, the whole sequence u_M^\dagger must converge to u^* in $H^t(\Omega)$. Since $t \in (2 + d/2, s)$, we also get pointwise convergence and convergence in $H^t(\Omega)$ for any $t < s$. ■

We note that this convergence theorem requires K to be adapted to the solution space of the PDE so that u^* belongs to \mathcal{U} . In our numerical experiments, we will use a Gaussian kernel. However, if f or the boundary $\partial\Omega$ are not sufficiently regular, then the embedding conditions $u^* \in \mathcal{U} \subseteq H^s(\Omega)$ may be satisfied by using kernel as the Green's function of some power of the Laplacian, instead of a Gaussian kernel; the latter induces smoothness on \mathcal{U} which may be incompatible with the regularity of u^* for irregular f and $\partial\Omega$. We will provide further insights into the convergence properties of our method, including an error bound between the solution u^* and a minimizer u^\dagger in Section 5.2.

1.1.4. Numerical framework

The representation of u^\dagger via the optimization problem (1.6) is the cornerstone of our numerical framework for solving nonlinear PDEs. Indeed, efficient solution of (1.6), and in turn construction and inversion of the matrix $K(\phi, \phi)$, are the most costly steps of our numerical implementation. We summarize several main ingredients of our algorithm below:

- We propose an efficient variant of the Gauss–Newton algorithm in Section 3.4.2. Although, in general, (1.6) is a nonconvex problem, our algorithm typically converges in between 2 and 10 steps in all the experiments we have conducted.

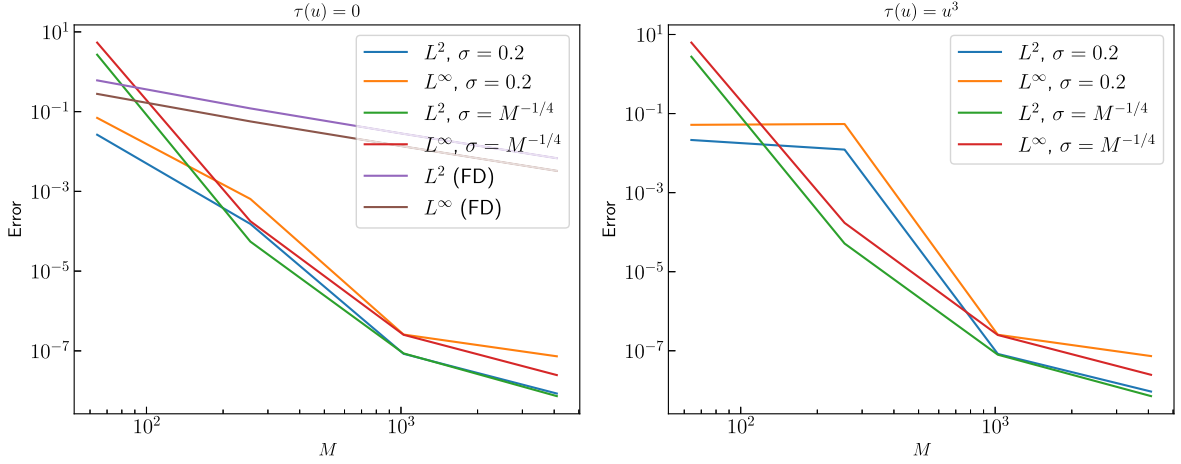


Fig. 1. L^2 and L^∞ error plots for numerical approximations of u^* , the solution to (1.1), as a function of the number of collocation points M . Left: $\tau(u) = 0$; both the kernel collocation method using Gaussian kernel with $\sigma = 0.2$ and $M^{-1/4}$ and the finite difference (FD) method were implemented. Right: $\tau(u) = u^3$; the kernel collocation method using Gaussian kernel with $\sigma = 0.2$ and $M^{-1/4}$ were used. In both cases, an adaptive nugget term with global parameter $\eta = 10^{-13}$ was used to regularize the kernel matrix Θ (see Appendix A.1 for details). (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

- In practice we perturb $K(\phi, \phi)$ to improve its condition number, and hence the numerical stability of the algorithm, by adding a small diagonal matrix; this perturbation is adapted to the problem at hand, as outlined in Appendix A.1; the approach generalizes the idea of a “nugget” as widely used in GP regression.
- To evaluate the cost function in (1.6), we pre-compute the Cholesky factorization of the (perturbed) kernel matrix and store it for multiple uses. State-of-the-art linear solvers for dense kernel matrices can be used for this step.

As a demonstration, we present here a simple experiment to show the convergence of our method. We take $d = 2$, $\Omega = (0, 1)^2$ and $\tau(u) = 0$ or u^3 together with homogeneous Dirichlet boundary conditions $g(\mathbf{x}) = 0$. The true solution is prescribed to be $u^*(\mathbf{x}) = \sin(\pi x_1) \sin(\pi x_2) + 4 \sin(4\pi x_1) \sin(4\pi x_2)$ and the corresponding right hand side $f(\mathbf{x})$ is computed using the equation.

We choose the Gaussian kernel

$$K(\mathbf{x}, \mathbf{y}; \sigma) = \exp\left(-\frac{|\mathbf{x} - \mathbf{y}|^2}{2\sigma^2}\right),$$

with lengthscale parameter σ ; we will fix this parameter, but note that the framework is naturally adapted to learning of such hyperparameters. We set $M = 64, 256, 1024, 4096$, sampling the collocation points on a uniform grid of points within Ω . We apply our algorithm to solve the PDEs and compute the L^2 and L^∞ errors to u^* . In the case $\tau(u) = 0$, which corresponds to a linear PDE, we also compare our algorithm with a second-order finite difference (FD) method. For the nonlinear case $\tau(u) = u^3$, we observe that the Gauss–Newton algorithm only needs 2 steps to converge. The errors versus M are depicted in Fig. 1. The following observations can be made from this figure:

- In the linear case $\tau(u) = 0$, where the corresponding optimization problem (1.6) is convex, our algorithm outperforms the FD method in terms of accuracy and rate of convergence. This can be attributed to the fact that the true solution is very smooth, and the Gaussian kernel allows for efficient approximation.
- The choice of the kernel parameter σ has a profound impact on the accuracy and rate of convergence of the algorithm, especially when M is not very large. This implies the importance of choosing a “good kernel” that is adapted to the solution space of the PDE, and highlights the importance of hyperparameter learning.
- In the nonlinear setting, our algorithm demonstrates similar convergence behavior to the linear setting. Once again, an appropriate choice of σ leads to significant gains in solution accuracy.

1.2. Relevant literature

Machine learning and statistical inference approaches to numerical approximation have attracted a lot of attention in recent years thanks to their remarkable success in engineering applications. Such approaches can be broadly divided into two categories: (1) GP/Kernel-based methods, and (2) ANN-based methods.

GPs and kernel-based methods have long been used in function approximation, regression, and machine learning [89]. As surveyed in [48]:

"[kernel approaches can be traced back to] Poincaré's course in Probability Theory [54] and to the pioneering investigations of Sul'din [75], Palasti and Renyi [52], Sard [63], Kimeldorf and Wahba [31] (on the correspondence between Bayesian estimation and spline smoothing/interpolation) and Larkin [35] (on the correspondence between Gaussian process regression and numerical approximation). Although their study initially attracted little attention among numerical analysts [35], it was revived in Information Based Complexity (IBC) [78], Bayesian numerical analysis [18], and more recently in Probabilistic Numerics [25] and Bayesian numerical homogenization [43]."

For recent reviews of the extensive applications of GP/kernel methods see [13,48,76] and [47, Chap. 20]. In particular, they have been introduced to motivate novel methods for solving ordinary differential equations (ODEs) [71,9,67] and underlie the collocation approach advocated for parameter identification in ODEs as described in [60]. For PDEs, the use of kernel methods can be traced back to meshless collocation methods with radial basis functions [91,88,64]. Furthermore, a recent systematic development towards solving PDEs was initiated in [43,44,51,47] and has lead to the identification of fast solvers for elliptic PDEs and dense kernel matrices [66,65] with state-of-the-art complexity versus accuracy guarantees. The effectiveness of these methods has been supported by well-developed theory [47] residing at the interfaces between numerical approximation [84,88], optimal recovery [40], information-based complexity [78], and GP regression [6], building on the perspective introduced in [18,35,40,79,84]. In particular there is a one to one correspondence [47,66] between (1) the locality of basis functions (gamblets) obtained with kernel methods and the *screening effect* observed in kriging [73], (2) Schur complementation and conditioning Gaussian vectors, and (3) the approximate low-rank structure of inverse stiffness matrices obtained with kernel methods and variational properties of Gaussian conditioning. Furthermore, although the approach of modeling a deterministic function (the solution u^* of a PDE) as a sample from a Gaussian distribution may seem counterintuitive, it can be understood (in the linear setting [47]) as an optimal minimax strategy for recovering u^* from partial measurements. Indeed, as in Von Neumann's theory of games, optimal strategies are mixed (randomized) strategies and (using quadratic norms to define losses) GP regression (kriging) emerges as an optimal minimax approach to numerical approximation [40,47].

On the other hand, ANN methods can be traced back to [19,33,34,61,80] and, although developed for ODEs several decades ago [11,61], with some of the work generalized to PDEs [32], their systematic development for solving PDEs has been initiated only recently. This systematic development includes the Deep Ritz Method [87], Physics Informed Neural Networks [58] (PINNs), DGN [70], and [23] which employs a probabilistic formulation of the nonlinear PDE via the Feynman-Kac formula. Although the basic idea is to replace unknown functions with ANNs and minimize some loss with respect to the parameters of the ANN to identify the solution, there are by now many variants of ANN approaches, and we refer to [29] for a recent review of this increasingly popular topic at the interface between scientific computing and deep learning. While ANN approaches have been shown to be empirically successful on complex problems (e.g., machine learning physics [59]), they may fail on simple ones [86,81] if the architecture of the ANN is not adapted to the underlying PDE [85]. Moreover, the theory of ANNs is still in its infancy; most ANN-based PDE solvers lack theoretical guarantees and convergence analyses are often limited to restricted linear instances [69,86]. Broadly speaking, in comparison with kernel methods, ANN methods have both limited theoretical foundations and unfavorable complexity vs accuracy estimates. We also remark that ANN methods are suitable for the learning of the parametric dependence of solutions of differential equations [92,5,36,37,7]; however, GP and kernel methods may also be used in this context, and the random feature method provides a natural approach to implementing such methods in high dimensions [41].

Regardless, the theory and computational framework of kernel methods may naturally be extended to ANN methods to investigate⁵ such methods and possibly accelerate them by viewing them as ridge regression with data-dependent kernels and following [66,65]. To this end, ANN methods can be interpreted as kernel methods with data-dependent kernels in two equivalent ways: (1) as solving PDEs in an RKHS space spanned by a feature map parameterized by the initial layers of the ANN that is learned from data; or, (2) as kernel-based methods with kernels that are parameterized by the inner layers of the network. For instance, [45] shows that Residual Neural Networks [24] (ResNets) are ridge regressors with warped kernels [62,53]. Since our approach employs an explicit kernel, it allows us to learn that kernel directly, as discussed in Subsection 5.3. Given the notorious difficulty of developing numerical methods for nonlinear PDEs [77], it is to some degree surprising that (as suggested by our framework) (A) this difficulty can universally be decomposed into three parts: (1) the compression/inversion of dense kernel matrices, (2) the selection of the kernel, and (3) the minimization of the reduced finite-dimensional optimization problem (3.10), and (B) a significant portion of the resulting analysis can be reduced to that of linear regression [47].

Beyond solving PDEs, ANN methods have also been used in data-driven discretizations, and discovery of PDEs that allow for the identification of the governing model [58,39,4]; this leads to applications in IPs. Our method, viewed as a GP conditioned on PDE constraints at collocation points, can be interpreted as solving an IP with Bayesian inference and a Gaussian prior [74]. Indeed, if we relax the PDE constraints as in Subsection 3.3.2 then a minimizer u^\dagger coincides with a MAP estimator of a posterior measure obtained by viewing the PDE constraints as nonlinear pointwise measurements of a field u with a Gaussian prior $\mathcal{N}(0, \mathcal{K})$. Bayesian IPs with Gaussian priors have been studied extensively (see [17,14,74] and references therein). The standard approach for their solution is to discretize the problem using spectral projection or

⁵ Beyond the infinite width neural tangent kernel regime [27,86].

finite element methods, and compute posterior MAP estimators [16] or employ Markov chain Monte Carlo algorithms [15] to simulate posterior samples. Our abstract representation theorem outlined in Section 2.1 completely characterizes the MAP estimator of posterior measures with Gaussian priors in settings where the forward map is written as the composition of a nonlinear map with bounded linear functionals acting on the parameter. Indeed, this is precisely the approach that we employ in Section 4 to solve IPs with PDE constraints. However, the main difference between our methodology and existing methods in the literature is that we pose the IP as that of recovering the solution of the PDE u^\dagger simultaneously with learning the unknown PDE parameter with independent Gaussian priors on both unknowns.

We now turn to motivation for the GP interpretation. The PDE solvers obtained here are deterministic and could be described from an entirely classical numerical approximation perspective. However we emphasize the GP interpretation for two reasons: (i) it is integral to the derivation of the methods; and (ii) it allows the numerical solution of the PDE to be integrated into a larger engineering pipeline and, in that context, the posterior distribution of the GP conditioned on the PDE at collocation points provides a measure of uncertainty quantification. Using the GP perspective as a pathway to the discovery of numerical methods, was the motivation for the work in [43,44]. Indeed, as discussed in [43], while the discovery of numerical solvers for PDEs is usually based on a combination of insight and guesswork, this process can be facilitated to the point of being automated, using this GP perspective. For instance, for nonsmooth PDEs, basis functions with near optimal accuracy/localization tradeoff and operator valued wavelets can be identified by conditioning physics informed Gaussian priors on localized linear measurements (e.g., local averages or pointwise values). Physics informed Gaussian priors can, in turn, be identified by (a) filtering uninformed Gaussian priors through the inverse operator [43], or (b) turning the process of computing fast with partial information into repeated zero-sum games with physics informed losses (whose optimal mixed strategies are physics informed Gaussian priors) [44]. The paper [57] generalized (a) to time-dependent PDEs by filtering uninformed priors through linearized PDEs obtained via time stepping. The paper [12] emphasized the probabilistic interpretation of numerical errors obtained from this Bayesian perspective. In particular [12, Sec. 5.2] describes a method identical to the one considered here (and [43]) for linear problems; in the setting of semi-linear PDEs, it is suggested in [12] that latent variables could be employed to efficiently sample from posterior/conditional measures (see also [45] where latent variables were employed to reduce nonlinear optimal recovery problems via two-level optimization as in (1.3)). Although the methodology proposed in our paper agrees with that in [12] for linear problems, the methodology we propose appears to be more general, and differs in the case of nonlinear problems to which both approaches apply.

1.3. Outline of article

The remainder of this article is organized as follows. We give an overview of the abstract theory of GPs on Banach spaces in Section 2; we establish notation, and summarize basic results and ideas that are used throughout the remainder of the article. Section 3 is dedicated to the development of our numerical framework for solving nonlinear PDEs with kernel methods; we outline our assumptions on the PDE, present a general convergence theory, discuss our approach to implementation, and present numerical experiments. In Section 4 we extend our nonlinear PDE framework to IPs and discuss the implementation differences between the PDE and IP settings, followed by numerical experiments involving a benchmark IP in subsurface flow. Finally, we present additional discussions, results, and possible extensions of our method in Section 5. Appendix A is devoted to the small diagonal regularization of kernel matrices (“nugget” term) and outlines general principles as well as specifics for the examples considered in this paper.

2. Conditioning GPs on nonlinear observations

In this section, we outline the abstract theory of RKHSs/GPs and their connection to Banach spaces endowed with quadratic norms; this forms the framework for the proposed methodology to solve PDEs and IPs. We start by recalling some basic facts about GPs in Subsection 2.1. This is followed in Subsection 2.2 by general results pertaining to conditioning of GPs on linear and nonlinear observations, leading to a representer theorem that is the cornerstone of our numerical algorithms. Some of these results may be known to experts, but to the best of our knowledge, they are not presented in the existing literature with the coherent goal of solving nonlinear problems via the conditioning of GPs; hence this material may be of independent interest to the reader.

2.1. GPs and Banach spaces endowed with a quadratic norm

Consider a separable Banach space \mathcal{U} and its dual \mathcal{U}^* with their duality pairing denoted by $[\cdot, \cdot]$. We further assume that \mathcal{U} is endowed with a quadratic norm $\|\cdot\|$, i.e., there exists a linear bijection $\mathcal{K} : \mathcal{U}^* \rightarrow \mathcal{U}$ that is symmetric ($[\mathcal{K}\phi, \varphi] = [\mathcal{K}\varphi, \phi]$), positive ($[\mathcal{K}\phi, \phi] > 0$ for $\phi \neq 0$), and such that

$$\|u\|^2 = [\mathcal{K}^{-1}u, u], \quad \forall u \in \mathcal{U}.$$

The operator \mathcal{K} endows \mathcal{U} and \mathcal{U}^* with the following inner products:

$$\begin{aligned} \langle u, v \rangle &:= [\mathcal{K}^{-1}u, v], \quad u, v \in \mathcal{U}, \\ \langle \phi, \varphi \rangle_* &:= [\phi, \mathcal{K}\varphi], \quad \phi, \varphi \in \mathcal{U}^*. \end{aligned}$$

Note that the $\langle \cdot, \cdot \rangle_*$ inner product defines a norm on \mathcal{U}^* that coincides with the standard dual norm of \mathcal{U}^* , i.e.,

$$\|\phi\|_* = \sup_{u \in \mathcal{U}, u \neq 0} \frac{[\phi, u]}{\|u\|} = \sqrt{[\phi, \mathcal{K}\phi]}.$$

As in [47, Chap. 11], although $\mathcal{U}, \mathcal{U}^*$ are also Hilbert spaces under the quadratic norms $\|\cdot\|$ and $\|\cdot\|_*$, we will keep using the Banach space terminology to emphasize the fact that our dual pairings will not be based on the inner product through the Riesz representation theorem, but on a different realization of the dual space. A particular case of the setting considered here is $\mathcal{U} = H_0^s(\Omega)$ (writing $H_0^s(\Omega)$ for the closure of the set of smooth functions with compact support in Ω with respect to the Sobolev norm $\|\cdot\|_{H^s(\Omega)}$), with its dual $\mathcal{U}^* = H^{-s}(\Omega)$ defined by the pairing $[\phi, v] := \int_{\Omega} \phi v$ obtained from the Gelfand triple $H^s(\Omega) \subset L^2(\Omega) \subset H^{-s}(\Omega)$. Here \mathcal{K} can be defined as solution map of an elliptic operator⁶ mapping $H_0^s(\Omega)$ to $H^{-s}(\Omega)$.

We say that ξ is the *canonical GP* [47, Chap. 17.6] on \mathcal{U} and write $\xi \sim \mathcal{N}(0, \mathcal{K})$, if and only if ξ is a linear isometry from \mathcal{U}^* to a centered Gaussian space (a closed linear space of scalar valued centered Gaussian random variables). The word canonical indicates that the covariance operator of ξ coincides with the bijection \mathcal{K} defining the norm on \mathcal{U} . Write $[\phi, \xi]$ for the image of $\phi \in \mathcal{U}^*$ under ξ and note that the following properties hold:

$$\mathbb{E}[\phi, \xi] = 0 \quad \text{and} \quad \mathbb{E}[\phi, \xi][\varphi, \xi] = [\phi, \mathcal{K}\varphi], \quad \forall \phi, \varphi \in \mathcal{U}^*.$$

The space \mathcal{U} coincides with the Cameron–Martin space of the GP $\mathcal{N}(0, \mathcal{K})$. In the setting where \mathcal{K} is defined through a covariance kernel K (such as in Subsection 1.1 and later in Subsection 3.4.2) then \mathcal{U} coincides with the RKHS space of the kernel K [82, Sec. 2.3].

2.2. Conditioning GPs with linear and nonlinear observations

Let ϕ_1, \dots, ϕ_N be N non-trivial elements of \mathcal{U}^* and define

$$\boldsymbol{\phi} := (\phi_1, \dots, \phi_N) \in (\mathcal{U}^*)^{\otimes N}. \quad (2.1)$$

Now consider the canonical GP $\xi \sim \mathcal{N}(0, \mathcal{K})$, then $[\boldsymbol{\phi}, \xi]$ is an \mathbb{R}^N -valued Gaussian vector and $[\boldsymbol{\phi}, \xi] \sim \mathcal{N}(0, \Theta)$ where

$$\Theta \in \mathbb{R}^{N \times N}, \quad \Theta_{i,n} = [\phi_i, \mathcal{K}\phi_n], \quad 1 \leq i, n \leq N. \quad (2.2)$$

The following proposition characterizes the conditional distribution of GPs under these linear observations; to simplify the statement it is useful to write $\mathcal{K}(\boldsymbol{\phi}, \varphi)$ for the vector with entries $[\phi_i, \mathcal{K}\varphi]$. This type of vectorized notation is used in [47].

Proposition 2.1. Consider a vector $\mathbf{y} \in \mathbb{R}^N$ and the canonical GP $\xi \sim \mathcal{N}(0, \mathcal{K})$. Then ξ conditioned on $[\boldsymbol{\phi}, \xi] = \mathbf{y}$ is also Gaussian. Moreover if Θ is invertible then $\text{Law}[\xi | [\boldsymbol{\phi}, \xi] = \mathbf{y}] = \mathcal{N}(u^\dagger, \mathcal{K}_\phi)$ with conditional mean defined by $u^\dagger = \mathbf{y}^T \Theta^{-1} \mathcal{K}\boldsymbol{\phi}$ and conditional covariance operator defined by $[\varphi, \mathcal{K}_\phi \varphi] = [\varphi, \mathcal{K}\varphi] - \mathcal{K}(\varphi, \boldsymbol{\phi}) \Theta^{-1} \mathcal{K}(\boldsymbol{\phi}, \varphi)$, $\forall \varphi \in \mathcal{U}^*$.

Proposition 2.1 gives a finite representation of the conditional mean of the GP constituting a representer theorem [47, Cor. 17.12]. Let us define the elements

$$\chi_i := \sum_{n=1}^N \Theta_{i,n}^{-1} \mathcal{K}\phi_n, \quad (2.3)$$

referred to as *gamblets* in the parlance of [44] which can equivalently be characterized as the minimizers of the variational problem

$$\begin{cases} \text{minimize} & \|\chi\| \\ \text{s.t.} & [\phi_n, \chi] = \delta_{i,n}, \quad n = 1, \dots, N. \end{cases} \quad (2.4)$$

This fact further enables the variational characterization of the conditional mean u^\dagger directly in terms of the gamblets χ_n .

Proposition 2.2. Let $u^\dagger = \mathbb{E}[\xi | [\boldsymbol{\phi}, \xi] = \mathbf{y}]$ as in Proposition 2.1. Then $u^\dagger = \sum_{n=1}^N y_n \chi_n$ is the unique minimizer of

$$\begin{cases} \text{minimize} & \|u\| \\ \text{s.t.} & [\phi_n, u] = y_n, \quad n = 1, \dots, N. \end{cases}$$

⁶ Given $s > 0$, we call an invertible operator $\mathcal{L} : H_0^s(\Omega) \rightarrow H^{-s}(\Omega)$ elliptic, if it is positive and symmetric in the sense that $\int_{\Omega} u \mathcal{L} u \geq 0$ and $\int_{\Omega} u \mathcal{L} v = \int_{\Omega} v \mathcal{L} u \geq 0$.

Proposition 2.2 is the cornerstone of our methodology for solution of nonlinear PDEs. It is also useful for the solution of IPs. For this purpose consider nonlinear functions $G: \mathbb{R}^N \rightarrow \mathbb{R}^I$ and $F: \mathbb{R}^N \rightarrow \mathbb{R}^M$ and vectors $\mathbf{o} \in \mathbb{R}^I$ and $\mathbf{y} \in \mathbb{R}^M$ and consider the optimization problem:

$$\begin{cases} \text{minimize}_{u \in \mathcal{U}} & \|u\|^2 + \frac{1}{\gamma^2} |G([\phi, u]) - \mathbf{o}|^2 \\ \text{s.t.} & F([\phi, u]) = \mathbf{y}, \end{cases} \quad (2.5)$$

where $\gamma \in \mathbb{R}$ is a parameter. We will use this formulation of IPs in PDEs, with u concatenating the solution of the forward PDE problem and the unknown parameter; the nonlinear constraint on F enforces the forward PDE and G the observed noisy data. Then a representer theorem still holds for a minimizer of this problem stating that the solution has a finite expansion in terms of the gamblers χ_n :

Proposition 2.3. Suppose $(\mathbf{o}, \mathbf{y}) \in \mathbb{R}^I \times \mathbb{R}^M$ are fixed and Θ is invertible.⁷ Then $u^\dagger \in \mathcal{U}$ is a minimizer of (2.5) if and only if $u^\dagger = \sum_{n=1}^N z_n^\dagger \chi_n$ and the vector \mathbf{z}^\dagger is a minimizer of

$$\begin{cases} \text{minimize}_{\mathbf{z} \in \mathbb{R}^N} & \mathbf{z}^T \Theta^{-1} \mathbf{z} + \frac{1}{\gamma^2} |G(\mathbf{z}) - \mathbf{o}|^2 \\ \text{s.t.} & F(\mathbf{z}) = \mathbf{y}. \end{cases} \quad (2.6)$$

Proof. The proof is nearly identical to the derivation of (1.5) presented in Section 1.1.2. Simply observe that minimizing (2.5) is equivalent to minimizing

$$\text{minimize}_{\mathbf{z} \in \mathbb{R}^N : F(\mathbf{z}) = \mathbf{y}} \begin{cases} \text{minimize}_{u \in \mathcal{U}} & \|u\|^2 + \frac{1}{\gamma^2} |G(\mathbf{z}) - \mathbf{o}|^2 \\ \text{s.t.} & [\phi, u] = \mathbf{z}. \end{cases} \quad (2.7)$$

Then solve the inner optimization problem for a fixed \mathbf{z} and apply Proposition 2.1. ■

We note that this model assumes independent and identically distributed (i.i.d.) observation noise for the vector \mathbf{o} and can easily be extended to correlated observation noise by replacing the misfit term $\frac{1}{\gamma^2} |G(\mathbf{z}) - \mathbf{o}|^2$ in (2.5) with an appropriately weighted misfit term of the form $|\Sigma^{-1/2}(G(\mathbf{z}) - \mathbf{o})|^2$ where Σ denotes the covariance matrix of the observation noise.

Remark 2.4. It is intuitive that a minimizer of the optimization problem we introduce and solve in this paper corresponds to a MAP point for the GP $\xi \sim \mathcal{N}(0, \mathcal{K})$ conditioned on PDE constraints at the collocation points. To prove this will require extension of the approach introduced in [16], for example, and is left for future work. Here we describe this connection informally in the absence of the equality constraints. Consider the prior measure $\mu_0 = \mathcal{N}(0, \mathcal{K})$ and consider the measurements $\mathbf{o} = G([\phi, u]) + \eta$, $\mathbf{y} = F([\phi, u]) + \eta'$, $\eta \sim \mathcal{N}(0, \gamma^2 I)$, $\eta' \sim \mathcal{N}(0, \beta^2 I)$. It then follows from Bayes' rule [74] that the posterior measure of u given the data (\mathbf{o}, \mathbf{y}) is identified as the measure

$$\begin{aligned} \frac{d\mu^{(\mathbf{o}, \mathbf{y})}}{d\mu_0}(u) &= \frac{1}{Z(\mathbf{o}, \mathbf{y})} \exp \left(-\frac{1}{2\gamma^2} |G([\phi, u]) - \mathbf{o}|^2 - \frac{1}{2\beta^2} |F([\phi, u]) - \mathbf{y}|^2 \right), \\ Z^{(\mathbf{o}, \mathbf{y})} &:= \mathbb{E}_{u \sim \mu_0} \exp \left(-\frac{1}{2\gamma^2} |G([\phi, u]) - \mathbf{o}|^2 - \frac{1}{2\beta^2} |F([\phi, u]) - \mathbf{y}|^2 \right). \end{aligned}$$

The article [16] showed that the MAP estimators of $\mu^{(\mathbf{o}, \mathbf{y})}$ are solutions to

$$\text{minimize}_{u \in \mathcal{U}} \quad \|u\|^2 + \frac{1}{\gamma^2} |G([\phi, u]) - \mathbf{o}|^2 + \frac{1}{\beta^2} |F([\phi, u]) - \mathbf{y}|^2.$$

Letting $\beta \rightarrow 0$ then yields (2.5). ◇

3. Solving nonlinear PDEs

In this section, we present our framework for solution of nonlinear PDEs by conditioning GPs on nonlinear constraints. In Subsection 3.1 we outline our abstract setting as well as our assumptions on PDEs of interest; this leads to Corollary 3.2

⁷ Relaxing the interpolation constraints renders the invertibility assumption on Θ unnecessary. Nevertheless we keep it for ease of presentation.

which states an analogue of Proposition 2.3 in the PDE setting. We analyze the convergence of our method in Subsection 3.2 and discuss two strategies for dealing with the nonlinear PDE constraints in Subsection 3.3. Next, we present the details pertaining to numerical implementations of our method, including the choice of kernels and a Gauss–Newton algorithm in Subsection 3.4. Finally, we present a set of numerical experiments in Subsection 3.5 that demonstrate the effectiveness of our method in the context of prototypical nonlinear PDEs.

3.1. Problem setup

Let us consider a bounded domain $\Omega \subseteq \mathbb{R}^d$ for $d \geq 1$ and a nonlinear PDE of the form

$$\begin{cases} \mathcal{P}(u^*)(\mathbf{x}) = f(\mathbf{x}), & \forall \mathbf{x} \in \Omega, \\ \mathcal{B}(u^*)(\mathbf{x}) = g(\mathbf{x}), & \forall \mathbf{x} \in \partial\Omega. \end{cases} \quad (3.1)$$

Here \mathcal{P} is a nonlinear differential operator and \mathcal{B} is an appropriate boundary operator with data f, g . Throughout this section and for brevity, we assume that the PDE at hand is well-defined pointwise and has a unique strong solution; extension of our methodology to weak solutions is left as a future research direction. We then consider \mathcal{U} to be an appropriate quadratic Banach space for the solution u^* such as a Sobolev space $H^s(\Omega)$ with sufficiently large regularity index $s > 0$.

We propose to solve the PDE (3.1) by approximating u^* by a GP conditioned on satisfying the PDE at a finite set of collocation points in $\overline{\Omega}$ and proceed to approximate the solution by computing the MAP point of such a conditioned GP. More precisely, let $\{\mathbf{x}_i\}_{i=1}^M$ be a collection of points in $\overline{\Omega}$ ordered in such a way that $\mathbf{x}_1, \dots, \mathbf{x}_{M_\Omega} \in \Omega$ are in the interior of Ω while $\mathbf{x}_{M_\Omega+1}, \dots, \mathbf{x}_M \in \partial\Omega$ are on the boundary. Now let \mathcal{U} be a quadratic Banach space with associated covariance operator $\mathcal{K} : \mathcal{U}^* \rightarrow \mathcal{U}$ and consider the optimization problem:

$$\begin{cases} \underset{u \in \mathcal{U}}{\text{minimize}} & \|u\| \\ \text{s.t.} & \mathcal{P}(u)(\mathbf{x}_m) = f(\mathbf{x}_m), \quad \text{for } m = 1, \dots, M, \\ & \mathcal{B}(u)(\mathbf{x}_m) = g(\mathbf{x}_m), \quad \text{for } m = M_\Omega + 1, \dots, M. \end{cases} \quad (3.2)$$

In other words, we wish to approximate u^* with the minimum norm element of the Cameron–Martin space of $\mathcal{N}(0, \mathcal{K})$ that satisfies the PDE and boundary data at the collocation points $\{\mathbf{x}_i\}_{i=1}^M$. In what follows we write $\mathcal{L}(\mathcal{U}; \mathcal{H})$ to denote the space of bounded and linear operators from \mathcal{U} to another Banach space \mathcal{H} . We make the following assumption regarding the operators \mathcal{P}, \mathcal{B} :

Assumption 3.1. There exist bounded and linear operators $L_1, \dots, L_Q \in \mathcal{L}(\mathcal{U}; C(\Omega))$ in which $L_1, \dots, L_{Q_b} \in \mathcal{L}(\mathcal{U}; C(\partial\Omega))$ for some $1 \leq Q_b \leq Q$, and there are maps $P : \mathbb{R}^Q \rightarrow \mathbb{R}$ and $B : \mathbb{R}^{Q_b} \rightarrow \mathbb{R}$, which may be nonlinear, so that $\mathcal{P}(u)(\mathbf{x})$ and $\mathcal{B}(u)(\mathbf{x})$ can be written as

$$\begin{aligned} \mathcal{P}(u)(\mathbf{x}) &= P(L_1(u)(\mathbf{x}), \dots, L_Q(u)(\mathbf{x})), & \forall \mathbf{x} \in \Omega, \\ \mathcal{B}(u)(\mathbf{x}) &= B(L_1(u)(\mathbf{x}), \dots, L_{Q_b}(u)(\mathbf{x})), & \forall \mathbf{x} \in \partial\Omega. \quad \diamond \end{aligned} \quad (3.3)$$

For prototypical nonlinear PDEs the L_q for $1 \leq q \leq Q$ are linear differential operators such as first or second order derivatives while the maps P and B are often simple algebraic nonlinearities. Furthermore, observe that for ease of presentation we are assuming fewer linear operators are used to define the boundary conditions than the operators that define the PDE in the interior.

Example NE (Nonlinear Elliptic PDE). Recall the nonlinear elliptic PDE (1.1) and consider the linear operators and nonlinear maps

$$L_1 : u \mapsto u, \quad L_2 : u \mapsto \Delta u, \quad P(v_1, v_2) = -v_2 + \tau(v_1), \quad B(v_1) = v_1,$$

where we took $Q = 2$ and $Q_b = 1$. Then this equation readily satisfies Assumption 3.1 whenever the solution is sufficiently regular so that $L_2(u)$ is well-defined pointwise within Ω . \diamond

Under Assumption 3.1 we can then define the functionals $\phi_m^{(q)} \in \mathcal{U}^*$ by setting

$$\phi_m^{(q)} := \delta_{\mathbf{x}_m} \circ L_q, \quad \text{where} \quad \begin{cases} 1 \leq m \leq M, & \text{if } 1 \leq q \leq Q_b, \\ 1 \leq m \leq M_\Omega, & \text{if } Q_{b+1} \leq q \leq Q. \end{cases} \quad (3.4)$$

We further use the shorthand notation $\phi^{(q)}$ to denote the vector of dual elements $\phi_m^{(q)}$ for a fixed index q . Observe that $\phi^{(q)} \in (\mathcal{U}^*)^{\otimes M}$ if $q \leq Q_b$ while $\phi^{(q)} \in (\mathcal{U}^*)^{\otimes M_\Omega}$ if $q > Q_b$. We further write

$$N = M Q_b + M_\Omega (Q - Q_b) \quad (3.5)$$

and define

$$\boldsymbol{\phi} := (\boldsymbol{\phi}^{(1)}, \dots, \boldsymbol{\phi}^{(Q)}) \in (\mathcal{U}^*)^{\otimes N}. \quad (3.6)$$

To this end, we define the measurement vector $\mathbf{y} \in \mathbb{R}^M$ by setting

$$y_m = \begin{cases} f(\mathbf{x}_m), & \text{if } m \in \{1, \dots, M_\Omega\}, \\ g(\mathbf{x}_m), & \text{if } m \in \{M_\Omega + 1, \dots, M\}, \end{cases} \quad (3.7)$$

as well as the nonlinear map

$$(F([\boldsymbol{\phi}, u]))_m := \begin{cases} P([\phi_m^{(1)}, u], \dots, [\phi_m^{(Q)}, u]) & \text{if } m \in \{1, \dots, M_\Omega\}, \\ B([\phi_m^{(1)}, u], \dots, [\phi_m^{(Q_b)}, u]) & \text{if } m \in \{M_\Omega + 1, \dots, M\}. \end{cases} \quad (3.8)$$

We can now rewrite the optimization problem (3.2) in the same form as (2.5):

$$\begin{cases} \text{minimize}_{u \in \mathcal{U}} & \|u\| \\ \text{s.t.} & F([\boldsymbol{\phi}, u]) = \mathbf{y}. \end{cases} \quad (3.9)$$

Then a direct application of Proposition 2.3 yields the following corollary.

Corollary 3.2. Suppose Assumption 3.1 holds, \mathcal{K} and Θ are invertible, and define $\boldsymbol{\phi}$, F , \mathbf{y} as above. Then u^\dagger is a minimizer of (3.2) if and only if $u^\dagger = \sum_{n=1}^N z_n^\dagger \chi_n$ where the χ_n are the gamblets defined according to (2.4) and \mathbf{z}^\dagger is a minimizer of

$$\begin{cases} \text{minimize}_{\mathbf{z} \in \mathbb{R}^N} & \mathbf{z}^T \Theta^{-1} \mathbf{z} \\ \text{s.t.} & F(\mathbf{z}) = \mathbf{y}. \end{cases} \quad (3.10)$$

The above corollary is the foundation of our numerical algorithms for approximation of the solution u^\dagger , as Θ^{-1} and the gamblets χ_n can be approximated offline while the coefficients z_n^\dagger can be computed by solving the optimization problem (3.10).

To solve (3.10) numerically, we will present two different approaches that transform it to an unconstrained optimization problem. Before moving to that in Subsection 3.3, we discuss the convergence theory first in the next section.

3.2. Convergence theory

We state and prove a more general version of Theorem 1.2 for our abstract setting of PDEs on Banach spaces with quadratic norms stating that a minimizer u^\dagger of (3.2) converges to the true solution u^* under sufficient regularity assumptions and for appropriate choices of the operator \mathcal{K} .

Theorem 3.3. Consider the PDE (3.1) and suppose that $\mathcal{U} \subset \mathcal{H} \subset C^t(\Omega) \cap C^{t'}(\overline{\Omega})$ where \mathcal{H} is a Banach space such that the first inclusion from the left is given by a compact embedding and $t \geq t' \geq 0$ are sufficiently large so that all derivatives appearing in the PDE are defined pointwise for elements of $C^t(\Omega) \cap C^{t'}(\overline{\Omega})$. Furthermore assume that the PDE has a unique classical solution $u^* \in \mathcal{U}$ and that, as $M \rightarrow \infty$,

$$\sup_{\mathbf{x} \in \Omega} \min_{1 \leq m \leq M_\Omega} |\mathbf{x} - \mathbf{x}_m| \rightarrow 0 \quad \text{and} \quad \sup_{\mathbf{x} \in \partial\Omega} \min_{M_\Omega+1 \leq m \leq M} |\mathbf{x} - \mathbf{x}_m| \rightarrow 0.$$

Write u_M^\dagger for a minimizer of (3.2) with M distinct collocation points. Then, as $M \rightarrow \infty$, the sequence of minimizers u_M^\dagger converges towards u^* pointwise in Ω and in \mathcal{H} .

Proof. The method of proof is similar to that of Theorem 1.2. Indeed, by the same argument as in the first paragraph of the proof for Theorem 1.2, there exists a subsequence $u_{M_p}^\dagger$ that converges to u_∞^\dagger in \mathcal{H} . This also implies convergence in $C^t(\Omega)$ and $C^{t'}(\overline{\Omega})$ due to the assumed continuous embedding of \mathcal{H} into $C^t(\Omega) \cap C^{t'}(\overline{\Omega})$. Since $t \geq t' \geq 0$ are sufficiently large so that all derivatives appearing in the PDE are defined pointwise for elements of $C^t(\Omega) \cap C^{t'}(\partial\Omega)$, we get that $\mathcal{P}u_{M_p}^\dagger$ converges to $\mathcal{P}u_\infty^\dagger$ in $C(\Omega)$ and $\mathcal{P}u_\infty^\dagger \in C(\Omega)$. As Ω is compact, u_∞^\dagger is also uniformly continuous in Ω .

For any $\mathbf{x} \in \Omega$ and $p \geq 1$, the triangle inequality shows that

$$\begin{aligned} |\mathcal{P}(u_\infty^\dagger)(\mathbf{x}) - f(\mathbf{x})| &\leq \min_{1 \leq m \leq M_{p,\Omega}} \left(|\mathcal{P}(u_\infty^\dagger)(\mathbf{x}) - \mathcal{P}(u_\infty^\dagger)(\mathbf{x}_m)| + |\mathcal{P}(u_\infty^\dagger)(\mathbf{x}_m) - \mathcal{P}(u_{M_p})(\mathbf{x}_m)| \right) \\ &\leq \min_{1 \leq m \leq M_{p,\Omega}} |\mathcal{P}(u_\infty^\dagger)(\mathbf{x}) - \mathcal{P}(u_\infty^\dagger)(\mathbf{x}_m)| + \|\mathcal{P}u_\infty^\dagger - \mathcal{P}u_{M_p}\|_{C(\Omega)}, \end{aligned} \quad (3.11)$$

where in the first inequality we have used the fact that $\mathcal{P}(u_{M_p})(\mathbf{x}_m) = f(\mathbf{x}_m)$. Here $M_{p,\Omega}$ is the number of interior points associated with the total M_p collocation points. Taking $p \rightarrow \infty$ and using the uniform continuity of $\mathcal{P}u_\infty^\dagger$ and the $C(\Omega)$ convergence from $\mathcal{P}u_{M_p}$ to $\mathcal{P}u_\infty^\dagger$, we derive that $\mathcal{P}(u_\infty^\dagger)(\mathbf{x}) = f(\mathbf{x})$. In a similar manner we can derive $\mathcal{B}(u_\infty^\dagger)(\mathbf{x}) = g(\mathbf{x})$. Thus, the limit u_∞^\dagger is a classical solution to the PDE. By the uniqueness of the solution we must have $u_\infty^\dagger = u^*$. Finally, as the limit u_∞^\dagger is independent of the choice of the subsequence, the whole sequence u_M^\dagger must converge to u^* pointwise and in \mathcal{H} . ■

We note that while this theorem does not provide a rate for convergence of u^\dagger towards u^* it relies on straightforward conditions that are readily verifiable for prototypical PDEs. Typically we choose $t, t' > 0$ large enough so that the PDE operators \mathcal{P}, \mathcal{B} are pointwise defined for the elements of $C^t(\Omega) \cap C^{t'}(\overline{\Omega})$ (e.g., $t > \text{order of PDE} + d/2$) and take the space \mathcal{H} to be a Sobolev-type space of appropriate regularity for the inclusion $\mathcal{H} \subset C^t(\Omega) \cap C^{t'}(\partial\Omega)$ to hold; also see the conditions of Theorem 1.2 and the subsequent discussion. The compact embedding $\mathcal{U} \subset \mathcal{H}$ can then be ensured by an appropriate choice of the covariance operator \mathcal{K} (or the associated kernel K). However, this choice should result in a sufficiently large space \mathcal{U} that includes the solution u^* of the PDE. Our conditions on the collocation points $\{\mathbf{x}_m\}_{m=1}^M$ simply ensure that these points form a dense subset of $\overline{\Omega}$ as $M \rightarrow \infty$.

3.3. Dealing with the constraints

Now, we turn our attention to the equality constraints in (3.10) and present two strategies for elimination or relaxation of these constraints; these transform the optimization problem to an unconstrained one. They are crucial preliminary steps before introducing our numerical framework.

3.3.1. Eliminating the equality constraints

The equality constraints in (3.10) can be eliminated under slightly stronger assumptions on the maps P, B . In particular, suppose that the following assumption holds:

Assumption 3.4. The equations

$$P(v_1, \dots, v_Q) = y, \quad B(v_1, \dots, v_{Q_b}) = y,$$

can be solved as finite-dimensional algebraic equations, i.e., there exist $\overline{P}: \mathbb{R}^{Q-1} \rightarrow \mathbb{R}$ and $\overline{B}: \mathbb{R}^{Q_b-1} \rightarrow \mathbb{R}$ so that

$$v_j = \overline{P}(v_1, \dots, v_{j-1}, v_{j+1}, \dots, v_Q, y), \quad v_k = \overline{B}(v_1, \dots, v_{k-1}, v_{k+1}, \dots, v_{Q_b}, y), \quad (3.12)$$

for selected indices $j \in \{1, \dots, Q\}$ and $k \in \{1, \dots, Q_b\}$. Then for integer N defined by (3.5), and using the solution maps $\overline{P}, \overline{B}$, we can then define a new solution map $\overline{F}: \mathbb{R}^{N-M} \times \mathbb{R}^M \rightarrow \mathbb{R}^N$ so that

$$F(\mathbf{z}) = \mathbf{y} \quad \text{if and only if} \quad \mathbf{z} = \overline{F}(\mathbf{w}, \mathbf{y}), \quad \text{for a unique } \mathbf{w} \in \mathbb{R}^{N-M}. \quad \diamond$$

With this new solution map we can rewrite (3.10) as an unconstrained optimization problem.

Corollary 3.5. Let Assumption 3.4 and the conditions of Corollary 3.2 hold. Then u^\dagger is a minimizer of (3.2) if and only if $u^\dagger = \sum_{n=1}^N z_n^\dagger \chi_n$ with $\mathbf{z}^\dagger = F'(\mathbf{w}^\dagger, \mathbf{y})$ and $\mathbf{w}^\dagger \in \mathbb{R}^{N-M}$ is a minimizer of

$$\underset{\mathbf{w} \in \mathbb{R}^{N-M}}{\text{minimize}} \quad \overline{F}(\mathbf{w}, \mathbf{y})^T \Theta^{-1} \overline{F}(\mathbf{w}, \mathbf{y}). \quad (3.13)$$

Example NE. Let us recall that we already eliminated the equality constraints in the context of the PDE (1.1) through the calculations leading to the unconstrained minimization problem (1.6). In that example, we used the calculation

$$P(v_1, v_2) = -v_2 + \tau(v_1) = y \Leftrightarrow v_2 = \tau(v_1) - y = \overline{P}(v_1, y),$$

that is, we solved Δu in terms of $\tau(u)$ and the source term in the interior of the domain in order to eliminate the PDE constraint. We further imposed the boundary conditions exactly since the boundary map B is simply the pointwise evaluation function in that example.

Alternatively, we could eliminate v_1 by setting $v_1 = \tau^{-1}(y + v_2)$, assuming that τ^{-1} has closed form. While both elimination strategies are conceptually valid they may lead to very different optimization problems. The former corresponds to solving for the values of u at the collocation points while the latter solves for the values of Δu at the interior points under Dirichlet boundary conditions at the boundary collocation points. \diamond

3.3.2. Relaxing the equality constraints

The choice of the solution maps \bar{P}, \bar{B} in (3.12), i.e., the choice of the variable which the equations are solved for, has an impact on the conditioning of (3.13); it is not a priori clear that poor conditioning can always be avoided by choice of variables to solve for. Moreover, for certain nonlinear PDEs Assumption 3.4 may not hold. In such cases it may be useful to relax the equality constraints in (3.9) and instead consider a loss of the following form:

$$\underset{u \in \mathcal{U}}{\text{minimize}} \quad \|u\|^2 + \frac{1}{\beta^2} |F([\phi, u]) - \mathbf{y}|^2, \quad (3.14)$$

where $\beta^2 > 0$ is a small positive parameter. Likewise (3.10) can be relaxed to obtain

$$\underset{\mathbf{z} \in \mathbb{R}^N}{\text{minimize}} \quad \mathbf{z}^T \Theta^{-1} \mathbf{z} + \frac{1}{\beta^2} |F(\mathbf{z}) - \mathbf{y}|^2. \quad (3.15)$$

Then a similar argument to the proof of Theorem 3.3 can be used to show that a minimizer of the relaxed optimization problem for u converges to the solution u^* of the PDE as the number of collocation points M increases and the parameter β vanishes.

Proposition 3.6. Fix $\beta > 0$. Then the optimization problem (3.14) has minimizer $u_{\beta, M}^\dagger$ which (assuming Θ to be invertible) may be expressed in the form

$$u_{\beta, M}^\dagger := \sum_{n=1}^N z_{\beta, n}^\dagger \chi_n \in \mathcal{U},$$

where z_{β}^\dagger denotes a minimizer of (3.15). Under the assumptions of Theorem 3.3 it follows that, as $(\beta, M^{-1}) \rightarrow 0$, the relaxed estimator $u_{\beta, M}^\dagger$ converges to u^* pointwise and in \mathcal{H} .

Proof. By the arguments used in Proposition 2.3 the minimizer of (3.15) delivers a minimizer of (3.14) in the desired form. Since u^* satisfies $F([\phi, u^*]) - \mathbf{y} = 0$ we must have $\|u_{\beta, M}^\dagger\| \leq \|u^*\|$. Then a compactness argument similar to that used in the proof of Theorem 3.3, noting that taking $\beta \rightarrow 0$ as $M \rightarrow \infty$ delivers exact satisfaction of the constraints in the limit, yields the desired result. \blacksquare

Example NE. When only part of the constraints $F(\mathbf{z}) = \mathbf{y}$ can be explicitly solved, as is often the case for boundary values, we can also combine the elimination and relaxation approach. Employing the relaxation approach for the interior constraint and the elimination approach for the boundary constraints in (1.1) amounts to replacing the optimization problem (1.5), which is the analogue of (3.10) for our running example, with the following problem for a small parameter $\beta^2 > 0$:

$$\begin{cases} \underset{\mathbf{z} \in \mathbb{R}^{M+M_\Omega}}{\text{minimize}} & \mathbf{z}^T K(\phi, \phi)^{-1} \mathbf{z} + \frac{1}{\beta^2} \left[\sum_{m=1}^{M_\Omega} |z_m^{(2)} + \tau(z_m^{(1)}) - f(\mathbf{x}_m)|^2 \right] \\ \text{s.t.} & z_m^{(1)} = g(\mathbf{x}_m), \quad \text{for } m = M_\Omega + 1, \dots, M. \end{cases} \quad (3.16)$$

We will numerically compare the above approach with the full elimination approach (1.6) in Subsection 3.5.1. \diamond

3.4. Implementation

We now outline the details of a numerical algorithm for solution of nonlinear PDEs based on the discussions of the previous subsection and in particular Corollary 3.2. We discuss the construction of the matrix Θ in Subsection 3.4.1 followed by a variant of the Gauss–Newton algorithm in Subsection 3.4.2 for solving the unconstrained or relaxed problems outlined in Subsections 3.3. We also note that strategies for regularizing the matrix Θ by adding small diagonal (“nugget”) terms are collected in Appendix A.

3.4.1. Constructing Θ

We established through Corollary 3.2 that a solution to (3.2) can be completely identified by \mathbf{z}^\dagger a minimizer of (3.10) as well as the gambles χ_n . Since here we are concerned with the strong form of the PDE (3.1) it is reasonable to assume that at the very least $\mathcal{U} \subset C(\bar{\Omega})$; although we often require higher regularity so that the PDE constraints can be imposed pointwise. This assumption suggests that our GP model for u^* can equivalently be identified via a covariance kernel function as opposed to the covariance operator \mathcal{K} . To this end, given a covariance operator \mathcal{K} define the covariance kernel K (equivalently Green's function of \mathcal{K}^{-1}) as

$$K : \bar{\Omega} \times \bar{\Omega} \mapsto \mathbb{R}, \quad K(\mathbf{x}, \mathbf{x}') := [\delta_{\mathbf{x}}, \mathcal{K} \delta_{\mathbf{x}'}]. \quad (3.17)$$

It is known that the kernel K completely characterizes the GP $\mathcal{N}(0, \mathcal{K})$ under mild conditions [82]; that is $\mathcal{N}(0, \mathcal{K}) \equiv \mathcal{GP}(0, K)$. Let us now consider the matrix Θ in block form

$$\Theta = \begin{bmatrix} \Theta^{(1,1)} & \Theta^{(1,2)} & \dots & \Theta^{(1,Q)} \\ \Theta^{(2,1)} & \Theta^{(2,2)} & \dots & \Theta^{(2,Q)} \\ \vdots & \vdots & \ddots & \vdots \\ \Theta^{(Q,1)} & \Theta^{(Q,2)} & \dots & \Theta^{(Q,Q)} \end{bmatrix}.$$

Using the $L^2(\Omega)$ duality pairing between \mathcal{U} and \mathcal{U}^* we can identify the blocks

$$\Theta^{(q,j)} = K(\phi^{(q)}, \phi^{(j)}),$$

where we used the shorthand notation of Subsection 1.1 for the kernel matrix, with the $\phi^{(q)}$ defined as in (3.4) and the subsequent discussion. To this end the entries of the $\Theta^{(q,j)}$ take the form

$$\Theta_{m,i}^{(q,j)} = L_q^{\mathbf{x}} L_j^{\mathbf{x}'} K(\mathbf{x}, \mathbf{x}')|_{(\mathbf{x}, \mathbf{x}') = (\mathbf{x}_m, \mathbf{x}_i)},$$

where we used the superscripts \mathbf{x}, \mathbf{x}' to denote the variables with respect to which the differential operators L_q, L_j act. Note that $\Theta \in \mathbb{R}^{N \times N}$ with $N = MQ_b + M_\Omega(Q - Q_b)$ following the definition of $\phi^{(q)}$ in Subsection 3.1.

3.4.2. A Gauss–Newton algorithm

Here we outline a variant of the Gauss–Newton algorithm [42, Sec. 10.3] for solution of the unconstrained optimization problem (3.13). Recall our definition of the maps \bar{P}, \bar{B} in (3.12) and in turn the map \bar{F} . We then propose to approximate a minimizer \mathbf{w}^\dagger of (3.13) with a sequence of elements \mathbf{w}^ℓ defined iteratively via $\mathbf{w}^{\ell+1} = \mathbf{w}^\ell + \alpha^\ell \delta \mathbf{w}^\ell$, where $\alpha^\ell > 0$ is an appropriate step size while $\delta \mathbf{w}^\ell$ is the minimizer of the optimization problem

$$\underset{\delta \mathbf{w} \in \mathbb{R}^{N-M}}{\text{minimize}} \quad \left(\bar{F}(\mathbf{w}^\ell, \mathbf{y}) + \delta \mathbf{w}^T \nabla \bar{F}(\mathbf{w}^\ell, \mathbf{y}) \right)^T \Theta^{-1} \left(\bar{F}(\mathbf{w}^\ell, \mathbf{y}) + \delta \mathbf{w}^T \nabla \bar{F}(\mathbf{w}^\ell, \mathbf{y}) \right),$$

and the gradient of \bar{F} is computed with respect to the \mathbf{w} variable only.⁸

This approach can be applied also to solve the relaxed problem (3.6) where this time we consider the sequence of approximations $\mathbf{z}^{\ell+1} = \mathbf{z}^\ell + \alpha^\ell \delta \mathbf{z}^\ell$, where $\delta \mathbf{z}^\ell$ is the minimizer of

$$\underset{\delta \mathbf{z} \in \mathbb{R}^N}{\text{minimize}} \quad \left(\mathbf{z}^\ell + \delta \mathbf{z} \right)^T \Theta^{-1} \left(\mathbf{z}^\ell + \delta \mathbf{z} \right) + \frac{1}{\beta^2} \left| F(\mathbf{z}^\ell) + \delta \mathbf{z}^T \nabla F(\mathbf{z}^\ell) - \mathbf{y} \right|^2.$$

Since (3.4.2) and (3.4.2) are both quadratic in $\delta \mathbf{w}$ and $\delta \mathbf{z}$ respectively, they can be solved exactly and efficiently at each step and the step-size parameters α^ℓ can be fixed or computed adaptively using standard step-size selection techniques [42]. However, in our experiments in Section 3.5, we find that both algorithms converge quickly simply by setting $\alpha^\ell = 1$.

Example NE. Let us return once more to the nonlinear elliptic PDE considered in Subsection 1.1. Observe that (1.6) is precisely in the form of (3.13) and so in order to formulate our Gauss–Newton iterations we need to linearize the vector valued function

$$\mathbf{w} \mapsto (\mathbf{w}, g(\mathbf{x}_{\partial\Omega}), f(\mathbf{x}_\Omega) - \tau(\mathbf{w})),$$

which can easily be achieved by linearizing τ . To this end, we solve (3.13) via the iteration $\mathbf{w}^{\ell+1} = \mathbf{w}^\ell + \alpha^\ell \delta \mathbf{w}^\ell$ where $\delta \mathbf{w}^\ell$ is the minimizer of the functional

⁸ Note that our proposed method is nothing more than the standard Gauss–Newton algorithm with Euclidean norm $|\cdot|$ defining the least-squares functional replaced with the weighted norm $|\Theta^{-1/2} \cdot|$ [42, Sec. 10.3].

$$(\mathbf{w}^\ell + \delta \mathbf{w}, g(\mathbf{x}_{\partial\Omega}), f(\mathbf{x}_\Omega) - \tau(\mathbf{w}^\ell) - \delta \mathbf{w}^T \nabla \tau(\mathbf{w}^\ell)) K(\phi, \phi)^{-1} \begin{pmatrix} \mathbf{w}^\ell + \delta \mathbf{w} \\ g(\mathbf{x}_{\partial\Omega}) \\ f(\mathbf{x}_\Omega) - \tau(\mathbf{w}^\ell) - \delta \mathbf{w}^T \nabla \tau(\mathbf{w}^\ell) \end{pmatrix}.$$

We also note that the sequence of approximations obtained by the above implementation of Gauss–Newton coincides with successive kernel collocation approximations of the solution of the following particular linearization of the PDE,

$$-\Delta u + u \tau'(u^n) = f - \tau(u^n) + u^n \tau'(u^n), \quad (3.18)$$

subject to the Dirichlet boundary conditions. \diamond

3.4.3. Computational bottlenecks

The primary computational cost of our method lies in the approximation of the matrix Θ^{-1} . Efficient factorizations and approximations of Θ^{-1} have been studied extensively in the GP regression literature [55] as well as spatial statistics, Kriging and numerical analysis (see [65,66] and the discussions within). In this paper, we do not employ these algorithms and choose instead to use standard $\mathcal{O}(N^3)$ algorithms to factorize Θ .

The algorithm introduced in [65] is particularly interesting as it directly approximates the Cholesky factors of Θ^{-1} by querying a subset of the entries of Θ . In fact, that algorithm alleviates the need for a small diagonal regularization (“nugget”) term by directly computing the Cholesky factors of Θ^{-1} from the entries of Θ . This could be done by extending the algorithm introduced and analyzed in [65]. This algorithm is based on the identification of an explicit formula for computing approximate Cholesky factors L minimizing the Kullback–Leibler divergence between $\mathcal{N}(0, \Theta^{-1})$ and $\mathcal{N}(0, LL^T)$ given a sparsity constraint on the entries of L . The proposed formula is equivalent to the Vecchia approximation [83] (popular in geostatistics). The resulting algorithm outlined in [65] computes ϵ approximate Cholesky factors of Θ^{-1} in $\mathcal{O}(N \log^{2d}(N/\epsilon))$ complexity by accessing $\mathcal{O}(N \log^d(N/\epsilon))$ entries of Θ .

Another possible bottleneck is the computation of the gambles χ_n . The articles [47,66] show that the gambles can be approximated with compactly supported functions in complexity $\mathcal{O}(N \log^{2d+1}(N/\epsilon))$. We also note that the complexity-vs-accuracy guarantees of [47,65,66] have only been established for functionals ϕ_n that are Dirac delta functions and kernels K that are the Green’s functions of arbitrary elliptic differential operators (mapping $H^s(\Omega)$ to $H^{-s}(\Omega)$). Extension of those results to functionals ϕ_n considered here is an interesting future direction.

3.5. Numerical experiments for nonlinear PDEs

In this subsection, we implement our algorithm to solve several nonlinear PDEs, including the nonlinear elliptic equation in Subsection 3.5.1, Burgers’ equation in Subsection 3.5.2 and the regularized Eikonal equation in Subsection 3.5.3. For all of these equations, we will start with a fixed M and demonstrate the performance of our algorithm by showing the pattern of collocation points, the loss function history of the Gauss–Newton iteration, and contours of the solution errors. Then, we vary the value of M and study how the errors change with respect to M . We also compare the elimination and relaxation approaches for dealing with the nonlinear constraints.

All the experiments are conducted using Python with the JAX package for automatic differentiation.⁹ In particular, we use automatic differentiation to form the kernel matrix Θ that involves derivatives of the kernel function, and to optimize the loss function via the Gauss–Newton method. Details on the choice of small diagonal regularization (“nugget”) terms for these experiments are presented in Appendices A.2 through A.4.

Remark 3.7. In all of the numerical experiments in this section we used a set of collocation points that are drawn randomly from the uniform distribution over the domain Ω , as opposed to the deterministic uniform grid used in Subsection 1.1.4. The choice of the random collocation points was made to highlight the flexibility of our methodology. Furthermore, random collocation points are often used in other machine learning algorithms for solution of PDEs such as PINNs [58] and so adopting this approach allows direct comparison with such methods. We observed empirically that the random grids had similar accuracy to the deterministic uniform grid in all experiments except for Burgers’ equation in Subsection 3.5.2, where random collocation points outperformed the uniform grid. Understanding this surprising performance gap is an interesting problem related to active learning and the acquisition of collocation points; we do not address this issue here. \diamond

Remark 3.8. Float64 data type was employed in the experiments below. This allows the use of small diagonal regularization (“nugget”) terms (see Appendix A for details) which do not affect accuracy in the computations described in this paper. In contrast, if Float32 data type (the default setting in JAX) is used, we found the need to regularize Θ with larger diagonal terms, leading to an observable accuracy floor. \diamond

⁹ We use JAX for convenience and all derivatives in our methodology can be computed using standard techniques such as symbolic computation or adjoint methods.

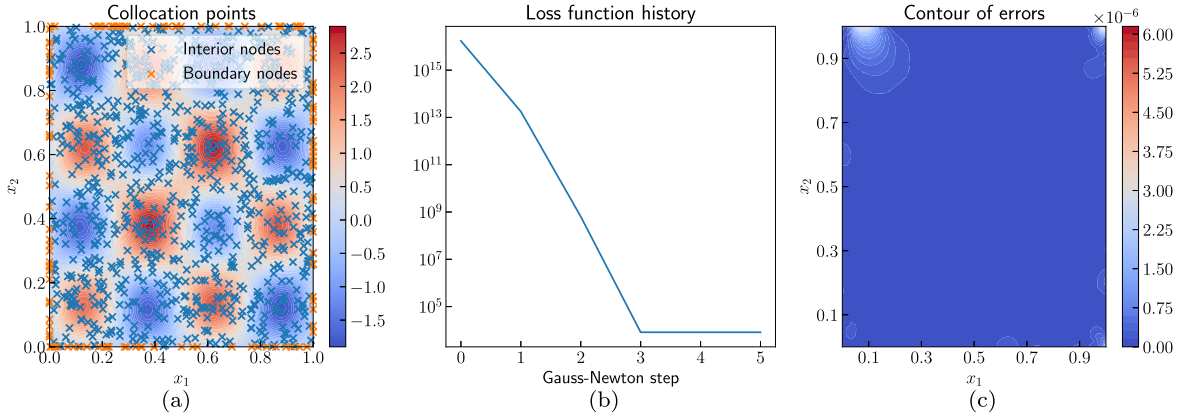


Fig. 2. Numerical results for the nonlinear elliptic PDE (1.1): (a) a sample of collocation points and contours of the true solution; (b) convergence history of the Gauss–Newton algorithm; (c) contours of the solution error. An adaptive nugget term with global parameter $\eta = 10^{-13}$ was employed (see Appendix A.2).

Table 1

Comparison between the elimination and relaxation approaches to deal with the equality constraints for the nonlinear elliptic PDE (1.1). Uniformly random collocation points were sampled with different M and $M_\Omega = 0.9M$. Adaptive nugget terms were employed with the global nugget parameter $\eta = 10^{-12}$ (see Appendix A.2). The lengthscale parameter $\sigma = 0.2$. Results were averaged over 10 realizations of the random collocation points. The maximum Gauss–Newton iteration was 10.

M	300	600	1200	2400
Elimination: L^2 error	4.84e-02	6.20e-05	2.74e-06	2.83e-07
Elimination: L^∞ error	3.78e-01	9.71e-04	4.56e-05	5.08e-06
Relaxation: L^2 error	1.15e-01	1.15e-04	1.87e-06	1.68e-07
Relaxation: L^∞ error	1.21e+00	1.45e-03	3.38e-05	1.84e-06

3.5.1. A nonlinear elliptic PDE

We revisit again the nonlinear elliptic equation in (1.1). As in Subsection 1.1.4, we take $d = 2$, $\Omega = (0, 1)^2$ and $\tau(u) = u^3$ together with homogeneous Dirichlet boundary conditions $g(\mathbf{x}) = 0$. The true solution is prescribed to be $u^*(\mathbf{x}) = \sin(\pi x_1) \sin(\pi x_2) + 4 \sin(4\pi x_1) \sin(4\pi x_2)$ and the corresponding right hand side $f(\mathbf{x})$ is computed using the equation. We choose the Gaussian kernel $K(\mathbf{x}, \mathbf{y}; \sigma) = \exp(-\frac{\|\mathbf{x} - \mathbf{y}\|^2}{2\sigma^2})$ with a lengthscale parameter σ .

First, for $M = 1024$ and $M_\Omega = 900$, we randomly sample collocation points in Ω as shown in Fig. 2(a). We further show an instance of the convergence history of the Gauss–Newton algorithm in Fig. 2(b) where we solved the unconstrained optimization problem (1.6) after eliminating the equality constraints. We used kernel parameter $\sigma = M^{-1/4}$ and appropriate nugget terms as outlined in Appendix A.2. We initiated the algorithm with a Gaussian random initial guess. We observe that only 3 steps sufficed for convergence. In Fig. 2(c), we show the contours of the solution error. The error in the approximate solution is seen to be fairly uniform spatially, with larger errors observed near the boundary, when $M = 1024$. We note that the main difference between these experiments and those in Subsection 1.1.4 is that here we used randomly distributed collocation points while a uniform grid was used previously.

Next, we compare two approaches for dealing with the PDE constraints as outlined in Subsection 3.3. We applied both the elimination and relaxation approaches, defined by the optimization problems (3.13) and (3.15) respectively, for different choices of M . In the relaxation approach, we set $\beta^2 = 10^{-10}$. Here we set $M = 300, 600, 1200, 2400$ and $M_\Omega = 0.9 \times M$. The L^2 and L^∞ errors of the converged Gauss–Newton solutions are shown in Table 1. Results were averaged over 10 realizations of the random collocation points. From the table we observe that the difference in solution errors was very mild and both methods were convergent as M increases. We note that in the relaxed setting, convergence is closely tied to our choice of β , and choosing an inadequate value, i.e. too small or too large, can lead to inaccurate solutions. In terms of computational costs, the elimination approaches take 2–3 steps of Gauss–Newton iterations on average, while the relaxation approach needs 5–8 steps. Thus while the elimination strategy appears to be more efficient, we do not observe a significant difference in the order of complexity of the methods for dealing with the constraints, especially when the number of collocation points becomes large.

Table 2

Space-time L^2 and L^∞ solution errors for the Burgers' equation (3.19) for different choices of M with kernel parameters $\sigma = (20, 3)$ and global nugget parameter $\eta = 10^{-5}$ if $M \leq 1200$ and $\eta = 10^{-10}$ otherwise (see Appendix A.3). Results were averaged over 10 realizations of the random collocation points. The maximum Gauss-Newton iteration was 30.

M	600	1200	2400	4800
L^2 error	1.75e-02	7.90e-03	8.65e-04	9.76e-05
L^∞ error	6.61e-01	6.39e-02	5.50e-03	7.36e-04

3.5.2. Burgers' equation

We consider numerical solution of the viscous Burgers equation:

$$\begin{aligned}\partial_t u + u \partial_s u - \nu \partial_s^2 u &= 0, \quad \forall (s, t) \in (-1, 1) \times (0, 1], \\ u(s, 0) &= -\sin(\pi x), \\ u(-1, t) &= u(1, t) = 0.\end{aligned}\tag{3.19}$$

We adopt an approach in which we solve the problem by conditioning a Gaussian process in space-time.¹⁰ In our experiments we take $\nu = 0.02$ and consider $\mathbf{x} = (s, t)$. We write this PDE in the form of (3.3) with $Q = 4$ and $Q_b = 1$ with linear operators $L_1(u) = u$, $L_2(u) = \partial_t u$, $L_3(u) = \partial_s u$, $L_4(u) = \partial_s^2 u$ and the nonlinear map $P(v_1, v_2, v_3, v_4) = v_2 + v_1 v_3 - \nu v_4^2$. The boundary part is simply $B(v_1) = v_1$. We then eliminate the equality constraints in our optimization framework following the approach of Subsection 3.3.1 using the equation $v_2 = \nu v_4^2 - v_1 v_3$.

We randomly sampled $M = 2400$ with $M_\Omega = 2000$ points in the computational domain $\Omega = [-1, 1] \times [0, 1]$ see Fig. 3(a), where we also plot contours of the true solution u . The Gauss-Newton algorithm was then applied to solve the unconstrained optimization problem. We computed the true solution from the Cole-Hopf transformation, together with the numerical quadrature. Since the time and space variability of the solution to Burgers' equation are significantly different, we chose an anisotropic kernel¹¹

$$K((s, t), (s', t'); \sigma) = \exp(-\sigma_1^{-2}(s - s')^2 - \sigma_2^{-2}(t - t')^2)$$

with $\sigma = (1/20, 1/3)$ together with an adaptive diagonal regularization ("nugget") as outlined in Appendix A.3.

We plot the Gauss-Newton iteration history in Fig. 3(b) and observe that 10 steps sufficed for convergence. We compare the converged solution to the true solution and present the contours of the error in Fig. 3(c). The maximum errors occurred close to the (viscous) shock at time 1 as expected. In Fig. 3(d-f), we also compare various time slices of the numerical and true solutions at times $t = 0.2, 0.5, 0.8$ to further highlight the ability of our method in capturing the location and shape of the shock.

Next, we studied the convergence properties of our method as a function of M as shown in Table 2. Here, we varied M with a fixed ratio of interior points, $M_\Omega/M = 5/6$. For each experiment we ran 10 steps of Gauss-Newton starting from a Gaussian random initial guess. Results were averaged over 10 realizations of the random collocation points. From the table, we observe that the error decreases very fast as M increases, implying the convergence of our proposed algorithm.

Finally, we note that the accuracy of our method is closely tied to the choice of the viscosity parameter ν and choosing a smaller value of ν , which in turn results in a sharper shock, can significantly reduce our accuracy. This phenomenon is not surprising since a sharper shock corresponds to the presence of shorter length and time scales in the solution; these in turn, require a more careful choice of the kernel, as well as suggesting the need to carefully choose the collocation points.

3.5.3. Eikonal PDE

We now consider the regularized Eikonal equation in $\Omega = [0, 1]^2$:

$$\begin{cases} |\nabla u(\mathbf{x})|^2 = f(\mathbf{x})^2 + \epsilon \Delta u(\mathbf{x}), & \forall \mathbf{x} \in \Omega, \\ u(\mathbf{x}) = 0, & \forall \mathbf{x} \in \partial\Omega, \end{cases}\tag{3.20}$$

with $f = 1$ and $\epsilon = 0.1$. We write this PDE in the form of (3.3) with $Q = 4$ and $Q_b = 1$ and linear operators $L_1(u) = u$, $L_2(u) = \partial_{x_1} u$, $L_3(u) = \partial_{x_2} u$, $L_4(u) = \Delta u$ and nonlinear map $P(v_1, v_2, v_3, v_4) = v_2^2 + v_3^2 - \epsilon v_4$ in the interior of Ω and define the boundary operator identically to Subsection 3.5.2. We further eliminate the nonlinear constraints, as outlined

¹⁰ It would also be possible to look at an incremental in time approach, for example using backward Euler discretization, in which one iteratively in time solves a nonlinear elliptic two point boundary value problem by conditioning a spatial Gaussian process; we do not pursue this here and leave it as a future direction.

¹¹ One can also design/learn a non-stationary kernel using the approaches discussed in Subsection 5.3. However, the parameterization of such kernels and strategies for tuning their hyperparameters are outside the scope of this article.

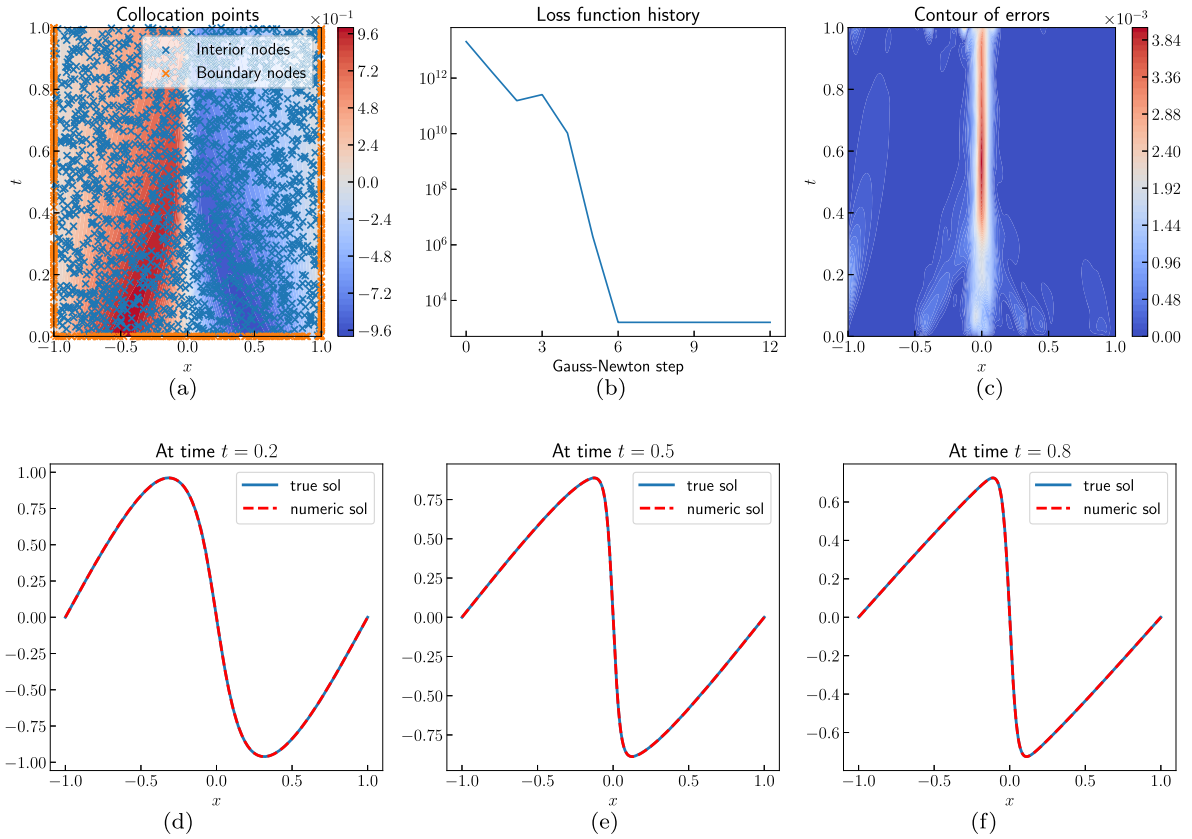


Fig. 3. Numerical results for Burgers equation (3.19): (a) an instance of uniformly sampled collocation points in space-time over contours of the true solution; (b) Gauss-Newton iteration history; (c) contours of the pointwise error of the numerical solution; (d–f) time slices of the numerical and true solutions at $t = 0.2, 0.5, 0.8$. An adaptive nugget term with global parameter $\eta = 10^{-10}$ was employed (see Appendix A.3).

in Subsection 3.3.1, by solving v_4 in terms of v_2, v_3 . To obtain a “true” solution, for the purpose of estimating errors, we employ the transformation $u = -\epsilon \log v$, which leads to the linear PDE $f v - \epsilon^2 \Delta v = 0$; we solve this by a highly-resolved FD method; we used 2000 uniform grid points in each dimension of the domain leading to the finest mesh that our hardware could handle.

As before, we began with $M = 2400$ collocation points with $M_\Omega = 2160$ interior points. An instance of these collocation points along with contours of the true solution are shown in Fig. 4(a). We employed a nugget term as outlined in Appendix A.4 and used the Gaussian kernel, as in Subsection 3.5.1 with $\sigma = M^{-1/4}$. Finally we used the Gauss-Newton algorithm to find the minimizer. We show the convergence history of Gauss-Newton in Fig. 4(b), observing that six iterations were sufficient for convergence. In Fig. 4(c) we show the error contours of the obtained numerical approximation, which appeared to be qualitatively different to Fig. 2(c) in that the errors were larger in the middle of the domain as well as close to the boundary.

Next we performed a convergence study by varying M and computing L^2 and L^∞ errors as reported in Table 3 by choosing the lengthscale parameter of the kernel $\sigma = M^{-1/4}$. We used the same nugget terms as in the Burgers’ equation (see Appendix A.4). Results were averaged over 10 realizations of the random collocation points. Once again we observe clear improvements in accuracy as the number of collocation points increases.

4. Solving inverse problems

We now turn our attention to solution of IPs and show that the methodology of Subsections 3.1–3.4 can readily be extended to solve such problems with small modifications. We describe the abstract setting of our IPs in Subsection 4.1 leading to Corollary 4.4 which is analogous to Proposition 2.3 and Corollary 3.2 in the setting of IPs. Subsection 4.2 outlines our approach for dealing with PDE constraints in IPs and highlights the differences in this setting in comparison to the PDE setting described in Subsection 3.3. Subsection 4.3 further highlights the implementation differences between the PDE and IP settings while Subsection 4.4 presents a numerical experiment concerning an IP in subsurface flow governed by the Darcy flow PDE.

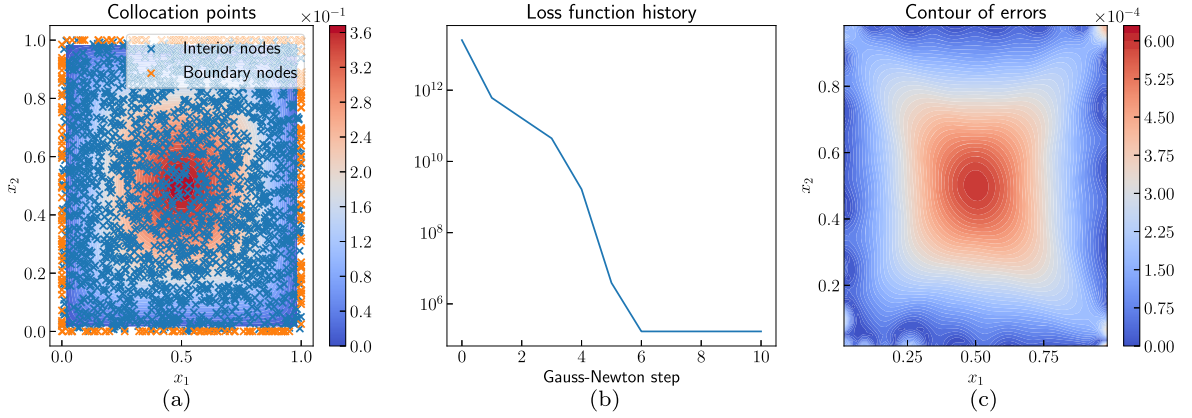


Fig. 4. Numerical results for the regularized Eikonal equation (3.20): (a) an instance of uniformly sampled collocation points over contours of the true solution; (b) Gauss–Newton iteration history; (c) contour of the solution error. An adaptive nugget term with $\eta = 10^{-10}$ was used (see Appendix A.4).

Table 3

Numerical results for the regularized Eikonal equation (3.20). Uniformly random collocation points were sampled with different M and with fixed ratio $M_\Omega = 0.9M$. An adaptive nugget term was used with global nugget parameter $\eta = 10^{-5}$ if $M \leq 1200$ and $\eta = 10^{-10}$ otherwise (see Appendix A.4), together with a Gaussian kernel with lengthscale parameter $\sigma = M^{-1/4}$. Results were averaged over 10 realizations of the random collocation points. The maximum Gauss–Newton iteration was 20.

M	300	600	1200	2400
L^2 error	1.01e-01	1.64e-02	2.27e-04	7.78e-05
L^∞ error	3.59e-01	7.76e-02	3.22e-03	1.61e-03

4.1. Problem setup

Consider our usual setting of a nonlinear parameteric PDE in strong form

$$\begin{cases} \mathcal{P}(u^*; a^*)(\mathbf{x}) = f(\mathbf{x}), & \forall \mathbf{x} \in \Omega, \\ \mathcal{B}(u^*; a^*)(\mathbf{x}) = g(\mathbf{x}), & \forall \mathbf{x} \in \partial\Omega. \end{cases} \quad (4.1)$$

As before we assume the solution u^* belongs to a quadratic Banach space \mathcal{U} while a^* is a parameter belonging to another quadratic Banach space \mathcal{A} . Our goal in this subsection is to identify the parameter a^* from limited observations of the solution u^* . To this end, fix $\psi_1, \dots, \psi_I \in \mathcal{U}^*$ and define

$$\boldsymbol{\psi} := (\psi_1, \dots, \psi_I) \in (\mathcal{U}^*)^{\otimes I}, \quad (4.2)$$

then our goal is to recover a^* given the noisy observations

$$\mathbf{o} = [\boldsymbol{\psi}, u] + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \gamma^2 I). \quad (4.3)$$

We propose to solve this inverse problem by modeling both u^* and a^* with canonical GPs on the spaces \mathcal{U}, \mathcal{A} with invertible covariance operators $\mathcal{K} : \mathcal{U}^* \rightarrow \mathcal{U}$ and $\tilde{\mathcal{K}} : \mathcal{A}^* \rightarrow \mathcal{A}$ respectively. We then condition these GPs to satisfy the PDE on collocation points $\mathbf{x}_1, \dots, \mathbf{x}_M \in \bar{\Omega}$ as before and propose to approximate u^*, a^* simultaneously via the optimization problem:

$$\begin{cases} \underset{(u,a) \in \mathcal{U} \times \mathcal{A}}{\text{minimize}} & \|u\|_{\mathcal{U}}^2 + \|a\|_{\mathcal{A}}^2 + \frac{1}{\gamma^2} \|\boldsymbol{\psi}, u - \mathbf{o}\|^2 \\ \text{s.t.} & \mathcal{P}(u; a)(\mathbf{x}_m) = f(\mathbf{x}_m), & \text{for } m = 1, \dots, M_\Omega, \\ & \mathcal{B}(u; a)(\mathbf{x}_m) = g(\mathbf{x}_m), & \text{for } m = M_\Omega + 1, \dots, M, \end{cases} \quad (4.4)$$

where we used subscripts to distinguish the quadratic norms on the spaces \mathcal{U} and \mathcal{A} .

Remark 4.1. In light of Remark 2.4 we observe that (4.4) corresponds to imposing a prior measure on u, a which assumes they are a priori independent. It is straightforward to introduce correlations between the solution u and the parameter a by defining the prior measure directly on the product space $\mathcal{U} \times \mathcal{A}$. This perspective will then lead to an analogous optimization problem to (4.4) with the same constraints but with the functional

$$\|(u, a)\|_{\mathcal{U} \times \mathcal{A}}^2 + \frac{1}{\gamma^2} |[\boldsymbol{\psi}, u] - \mathbf{o}|^2,$$

where we used $\|\cdot\|_{\mathcal{U} \times \mathcal{A}}$ to denote the RKHS norm of the GP associated with $\mathcal{U} \times \mathcal{A}$. \diamond

Remark 4.2. We also note that the Bayesian interpretation of (4.4) can be viewed as an extension of gradient matching [9,38] from ODEs to PDEs. Indeed, gradient matching simultaneously approximates the unknown parameters and the solution of an ODE system using a joint GP prior and imposes the ODE as a constraint at finitely many time steps. \diamond

We make analogous assumptions on the form of the operators \mathcal{P}, \mathcal{B} as in Assumption 3.1 but this time also involving the parameters a :

Assumption 4.3. There exist bounded and linear operators $L_1, \dots, L_Q \in \mathcal{L}(\mathcal{U}; C(\Omega))$ in which $L_1, \dots, L_{Q_b} \in \mathcal{L}(\mathcal{U}; C(\partial\Omega))$ for some $1 \leq Q_b \leq Q$, and $\tilde{L}_1, \dots, \tilde{L}_J \in \mathcal{L}(\mathcal{A}; C(\bar{\Omega}))$ together with maps $P : \mathbb{R}^{Q+J} \rightarrow \mathbb{R}$ and $B : \mathbb{R}^{Q_b+J} \rightarrow \mathbb{R}$, which may be nonlinear, so that $\mathcal{P}(u; a)(\mathbf{x})$ and $\mathcal{B}(u; a)(\mathbf{x})$ can be written as

$$\begin{aligned} \mathcal{P}(u; a)(\mathbf{x}) &= P(L_1(u)(\mathbf{x}), \dots, L_Q(u)(\mathbf{x}); \tilde{L}_1(a)(\mathbf{x}), \dots, \tilde{L}_J(a)(\mathbf{x})), \quad \forall \mathbf{x} \in \Omega, \\ \mathcal{B}(u; a)(\mathbf{x}) &= B(L_1(u)(\mathbf{x}), \dots, L_{Q_b}(u)(\mathbf{x}); \tilde{L}_1(a)(\mathbf{x}), \dots, \tilde{L}_J(a)(\mathbf{x})), \quad \forall \mathbf{x} \in \partial\Omega. \quad \diamond \end{aligned} \quad (4.5)$$

Similarly to the L_q , the \tilde{L}_j operators are also linear differential operators in case of prototypical PDEs while the maps P, B remain as simple algebraic nonlinearities. Let us briefly consider an IP in subsurface flow and verify the above assumption.

Example DF (Darcy flow IP). Let $\Omega = (0, 1)^2$ and consider the Darcy flow PDE with Dirichlet boundary conditions

$$\begin{cases} -\operatorname{div}(\exp(a)\nabla u)(\mathbf{x}) = f(\mathbf{x}), & \mathbf{x} \in \Omega, \\ u(\mathbf{x}) = 0, & \mathbf{x} \in \partial\Omega. \end{cases}$$

We wish to approximate $a \in C^1(\bar{\Omega})$ given noisy pointwise observations of u at a set of points $\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_J$.¹² Thus, we take $\psi_i = \delta_{\tilde{\mathbf{x}}_i}$. By expanding the PDE we obtain

$$-\operatorname{div}(\exp(a)\nabla u) = -\exp(a)(\nabla a \cdot \nabla u + \Delta u),$$

and so we simply choose $Q = 3$, $Q_b = 1$ and $J = 2$ with the linear operators

$$L_1(u) = u, \quad L_2(u) = \nabla u, \quad L_3(u) = \Delta u, \quad \tilde{L}_1(a) = a, \quad \tilde{L}_2(a) = \nabla a.$$

We can then satisfy Assumption (4.5) by taking

$$P(v_1, v_2, v_3; v_4, v_5) = -\exp(v_4)(v_5 \cdot v_2 + v_3), \quad B(v_1; v_4, v_5) = v_1, \quad (4.6)$$

where we have slightly abused notation by letting L_2, \tilde{L}_2 be vector valued and defining P, B to take vectors as some of their inputs. \diamond

As in Subsection 3.1 we now define the functionals $\phi_m^{(q)} \in \mathcal{U}^*$ for $m = 1, \dots, M$ and $q = 1, \dots, Q$ according to (3.4) and (3.6) with $N = MQ_b + M_\Omega(Q - Q_b)$. Similarly we define the functionals $\tilde{\phi}_m^{(j)} \in \mathcal{A}^*$ as

$$\tilde{\phi}_m^{(j)} := \delta_{\tilde{\mathbf{x}}_m} \circ \tilde{L}_j, \quad \text{for } m = 1, \dots, M, \text{ and } j = 1, \dots, J, \quad (4.7)$$

together with the vectors

$$\tilde{\boldsymbol{\phi}}^{(j)} = (\tilde{\phi}_1^{(j)}, \dots, \tilde{\phi}_M^{(j)}) \in (\mathcal{A}^*)^{\otimes M} \quad \text{and} \quad \tilde{\boldsymbol{\phi}} = (\tilde{\boldsymbol{\phi}}^{(1)}, \dots, \tilde{\boldsymbol{\phi}}^{(J)}) \in (\mathcal{A}^*)^{\otimes \tilde{N}}, \quad (4.8)$$

where $\tilde{N} := MJ$. Similarly to (3.8) define the map

$$\begin{aligned} (F([\boldsymbol{\phi}, u]_{\mathcal{U}}; [\tilde{\boldsymbol{\phi}}, a]_{\mathcal{A}}))_m &:= \\ &\begin{cases} P([\phi_m^{(1)}, u]_{\mathcal{U}}, \dots, [\phi_m^{(Q)}, u]_{\mathcal{U}}; [\tilde{\phi}_m^{(1)}, a]_{\mathcal{A}}, \dots, [\tilde{\phi}_m^{(J)}, a]_{\mathcal{A}}) & \text{if } m \in \{1, \dots, M_\Omega\}, \\ B([\phi_m^{(1)}, u]_{\mathcal{U}}, \dots, [\phi_m^{(Q_b)}, u]_{\mathcal{U}}; [\tilde{\phi}_m^{(1)}, a]_{\mathcal{A}}, \dots, [\tilde{\phi}_m^{(J)}, a]_{\mathcal{A}}) & \text{if } m \in \{M_\Omega + 1, \dots, M\}, \end{cases} \end{aligned}$$

where we used subscripts to distinguish the duality pairings between $\mathcal{U}, \mathcal{U}^*$ and the pairing between $\mathcal{A}, \mathcal{A}^*$. With this new notation we can finally rewrite (4.4) in the familiar form

¹² We note that while the Darcy flow PDE is linear in u the mapping from a to u is nonlinear resulting in a nonlinear inverse problem.

$$\begin{cases} \text{minimize}_{(u,a) \in \mathcal{U} \times \mathcal{A}} & \|u\|_{\mathcal{U}}^2 + \|a\|_{\mathcal{A}}^2 + \frac{1}{\gamma^2} |\psi(u) - \mathbf{o}|^2 \\ \text{s.t.} & F([\phi, u]_{\mathcal{U}}; [\tilde{\phi}, a]_{\mathcal{A}}) = \mathbf{y}, \end{cases} \quad (4.9)$$

with the PDE data vector $\mathbf{y} \in \mathbb{R}^M$ defined in (3.7).

We can now apply Proposition 2.3 with the canonical GP defined on the product space $\mathcal{U} \times \mathcal{A}$ and with a block diagonal covariance operator $\mathcal{K} \otimes \tilde{\mathcal{K}}$ to obtain a representer theorem for minimizer of (4.9). We state this result as a corollary below after introducing some further notation. Define the vector $\varphi = (\varphi_1, \dots, \varphi_{N+I}) \in (\mathcal{U}^*)^{\otimes(I+N)}$, with entries¹³

$$\varphi_n := \begin{cases} \psi_n, & \text{if } n = 1, \dots, I, \\ \phi_{n-I}, & \text{if } n = I+1, \dots, I+N, \end{cases}$$

as well as the matrices $\Theta \in \mathbb{R}^{(I+N) \times (I+N)}$ and $\tilde{\Theta} \in \mathbb{R}^{\tilde{N} \times \tilde{N}}$ with entries

$$\Theta_{i,n} = [\varphi_i, \mathcal{K}\varphi_n]_{\mathcal{U}} \quad \text{and} \quad \tilde{\Theta}_{i,n} = [\tilde{\phi}_i, \tilde{\mathcal{K}}\tilde{\phi}_n]_{\mathcal{A}}.$$

As in (2.3) we define the gamblets

$$\chi_i = \sum_{n=1}^{N+I} \Theta_{i,n}^{-1} \mathcal{K}\varphi_n, \quad \text{and} \quad \tilde{\chi}_i = \sum_{n=1}^{\tilde{N}} \tilde{\Theta}_{i,n}^{-1} \tilde{\mathcal{K}}\tilde{\phi}_n.$$

Then Proposition 2.3 yields the following corollary.

Corollary 4.4. Suppose Assumption 4.3 holds and that the covariance operators \mathcal{K} and $\tilde{\mathcal{K}}$ as well as the matrices Θ and $\tilde{\Theta}$ are invertible. Then $(u^\dagger, a^\dagger) \in \mathcal{U} \times \mathcal{A}$ is a minimizer of (3.2) if and only if

$$u^\dagger = \sum_{n=1}^{I+N} z_n^\dagger \chi_n, \quad \text{and} \quad a^\dagger = \sum_{n=1}^{\tilde{N}} \tilde{z}_n^\dagger \tilde{\chi}_n,$$

where the vectors $\mathbf{z}^\dagger, \tilde{\mathbf{z}}^\dagger$ are minimizers of

$$\begin{cases} \text{minimize}_{(\mathbf{z}, \tilde{\mathbf{z}}) \in (\mathbb{R}^{I+N} \times \mathbb{R}^{\tilde{N}})} & \mathbf{z}^T \Theta^{-1} \mathbf{z} + \tilde{\mathbf{z}}^T \tilde{\Theta}^{-1} \tilde{\mathbf{z}} + \frac{1}{\gamma^2} |\Pi^I \mathbf{z} - \mathbf{o}|^2 \\ \text{s.t.} & F(\Pi_N \mathbf{z}; \tilde{\mathbf{z}}) = \mathbf{y}, \end{cases} \quad (4.10)$$

where $\Pi^I : \mathbb{R}^{I+N} \rightarrow \mathbb{R}^I$ is the projection that extracts the first I entries of a vector while $\Pi_N : \mathbb{R}^{I+N} \rightarrow \mathbb{R}^N$ is the projection that extracts the last N entries.

4.2. Dealing with the constraints

The equality constraints in (4.10) can be dealt with using the same strategies as in Subsection 3.3. Indeed, as in Subsection 3.3.2, we can readily relax these constraints to obtain the optimization problem

$$\text{minimize}_{(\mathbf{z}, \tilde{\mathbf{z}}) \in (\mathbb{R}^{I+N} \times \mathbb{R}^{\tilde{N}})} \quad \mathbf{z}^T \Theta^{-1} \mathbf{z} + \tilde{\mathbf{z}}^T \tilde{\Theta}^{-1} \tilde{\mathbf{z}} + \frac{1}{\gamma^2} |\Pi^I \mathbf{z} - \mathbf{o}|^2 + \frac{1}{\beta^2} |F(\Pi_N \mathbf{z}; \tilde{\mathbf{z}}) - \mathbf{y}|^2, \quad (4.11)$$

for a small parameter $\beta^2 > 0$. Elimination of the constraints as in Subsection 3.3.1 is slightly more delicate, but is sometimes possible. Suppose there exists a solution map $\bar{F} : \mathbb{R}^{N+\tilde{N}-M} \times \mathbb{R}^M \rightarrow \mathbb{R}^{N+\tilde{N}}$ so that

$$F(\Pi_N \mathbf{z}; \tilde{\mathbf{z}}) = \mathbf{y} \quad \text{if and only if} \quad (\Pi_N \mathbf{z}, \tilde{\mathbf{z}}) = \bar{F}(\mathbf{w}, \mathbf{y}) \quad \text{for a unique } \mathbf{w} \in \mathbb{R}^{N+\tilde{N}-M}.$$

Then solving (4.10) is equivalent to solving the unconstrained problem

$$\text{minimize}_{(\mathbf{v}, \mathbf{w}) \in \mathbb{R}^I \times \mathbb{R}^{N+\tilde{N}-M}} \quad (\mathbf{v}, \bar{F}(\mathbf{w}, \mathbf{y})) \begin{bmatrix} \Theta^{-1} & 0 \\ 0 & \tilde{\Theta}^{-1} \end{bmatrix} \begin{pmatrix} \mathbf{v} \\ \bar{F}(\mathbf{w}, \mathbf{y}) \end{pmatrix} + \frac{1}{\gamma^2} |\mathbf{v} - \mathbf{o}|^2, \quad (4.12)$$

and setting $\Pi^I \mathbf{z}^\dagger = \mathbf{v}^\dagger$ and $(\Pi_N \mathbf{z}^\dagger, \tilde{\mathbf{z}}) = \bar{F}(\mathbf{w}^\dagger, \mathbf{y})$.

¹³ Note that we are concatenating the I measurement functionals defining the data for the IP with the N functionals used to define the PDE at the collocation points.

4.3. Implementation

Both of the problems (4.11) and (4.12) can be solved using the same techniques outlined in Subsection 3.4 except that we now have a higher dimensional solution space. Below we briefly describe the main differences between the implementation of the PDE and IP solvers.

4.3.1. Constructing Θ and $\tilde{\Theta}$

We propose to construct the matrices Θ , $\tilde{\Theta}$ using appropriate kernels K , for the solution u of the PDE, and \tilde{K} , for the parameter a identically to Subsection 3.4.1. Our minimum requirements on K , \tilde{K} is sufficient regularity for the pointwise constraints in (4.4) to be well-defined. Since we have limited and noisy data in the inverse problem setting, it is not possible for us to recover the exact solution (u^*, a^*) in general and so the kernels K , \tilde{K} should be chosen to reflect our prior assumptions on the unknown parameter and the solution of the PDE at that parameter value.

4.4. Numerical experiments for Darcy flow

In this subsection, we apply our method to an IP involving the Darcy flow PDE. We consider the IP outlined in Example DF with the true coefficient $a^*(\mathbf{x})$ satisfying

$$\exp(a^*(\mathbf{x})) = \exp(\sin(2\pi x_1) + \sin(2\pi x_2)) + \exp(-\sin(2\pi x_1) - \sin(2\pi x_2)),$$

and the right hand side source term is $f = 1$. We randomly sampled $M = 500$ collocation points with $M_\Omega = 400$ in the interior. From these 400 interior points, we randomly chose $I = 40$ points and observed the values of $u(\mathbf{x})$ at those points as the data for the IP. The values of $u(\mathbf{x})$ for this purpose were generated by first solving the equation with the true coefficient on a uniform grid and then using linear interpolation to get the solution at the observation points. We further added independent Gaussian noise $\mathcal{N}(0, \gamma^2 I)$ with noise standard deviation $\gamma = 10^{-3}$ to these observations. In dealing with the nonlinear constraint shown in Example DF, we eliminated the variable v_3 using the relation in (4.6).

We chose Gaussian kernels for both u and a with the same lengthscale parameter $\sigma = 0.2$ and adaptive diagonal (“nugget”) terms were added to the kernel matrices, with parameters $\eta = \tilde{\eta} = 10^{-5}$, to regularize Θ and $\tilde{\Theta}$ as outlined in Appendix A.5. In Fig. 5 we show the experimental results for recovering both a and u . From the figure, we observe that the Gauss–Newton iterations converged after 6 steps. Moreover, the recovered a and u are reasonably accurate, i.e. they capture the shape of the truth, given the limited amount of observation information available.

5. Concluding remarks

We have introduced a kernel/GP framework for solving nonlinear PDEs and IPs centered around the idea of approximating the solution of a given PDE with a MAP estimator of a GP conditioned on satisfying the PDE at a set of collocation points. Theoretically, we exhibited a nonlinear representer theorem which finite-dimensionalizes the MAP estimation problem and proved the convergence of the resulting solution towards the truth as the number of collocation points goes to infinity, under some regularity assumptions. Computationally, we demonstrated that the solution can be found by solving a finite-dimensional optimization problem with quadratic loss and nonlinear constraints. We presented two methods for dealing with the nonlinear constraints, namely the elimination and relaxation approaches. An efficient variant of the Gauss–Newton algorithm was also proposed for solving the resulting unconstrained optimization problem, where an adaptive nugget term was employed for regularization together with offline Cholesky factorizations of the underlying kernel matrices. We demonstrated that the proposed algorithm performs well in a wide range of prototypical nonlinear problems such as a nonlinear elliptic PDE, Burgers’ equation, a regularized Eikonal equation, and the identification of the permeability field in Darcy flow.

While our theoretical results established the existence of a solution for the finite dimensional optimization problem and the convergence of u^\dagger to the truth, the uniqueness of the solution is not guaranteed and the convergence theorem does not imply convergence rates. In what follows, we provide several additional discussions that hold the potential to address these two issues and point towards interesting future directions: in Subsection 5.1 we discuss how to get the uniqueness result if an appropriate condition is assumed. In Subsection 5.2, we lay a path to obtain rigorous error estimates followed by a discussion about learning of hierarchical kernel parameters in Subsection 5.3. Finally, in Subsection 5.4, we connect the framework of this article to MAP estimation for Bayesian IPs with non-Gaussian priors.

5.1. Uniqueness and optimality of minimizers

While our theoretical results in Subsection 2.2 characterize the minimizers of nonlinearly constrained optimization problems in RKHSs they do not imply the uniqueness of such minimizers. The lack of convexity due to the nonlinearity of F makes it impossible to obtain such uniqueness results, however, a positive result can be obtained under appropriate assumptions on the function F . Recall (3.10), and assume that the following condition holds

$$(\mathbf{z} - \mathbf{z}^\dagger)^T \Theta^{-1} \mathbf{z}^\dagger \geq 0 \text{ for } F(\mathbf{z}) = \mathbf{y}. \quad (5.1)$$

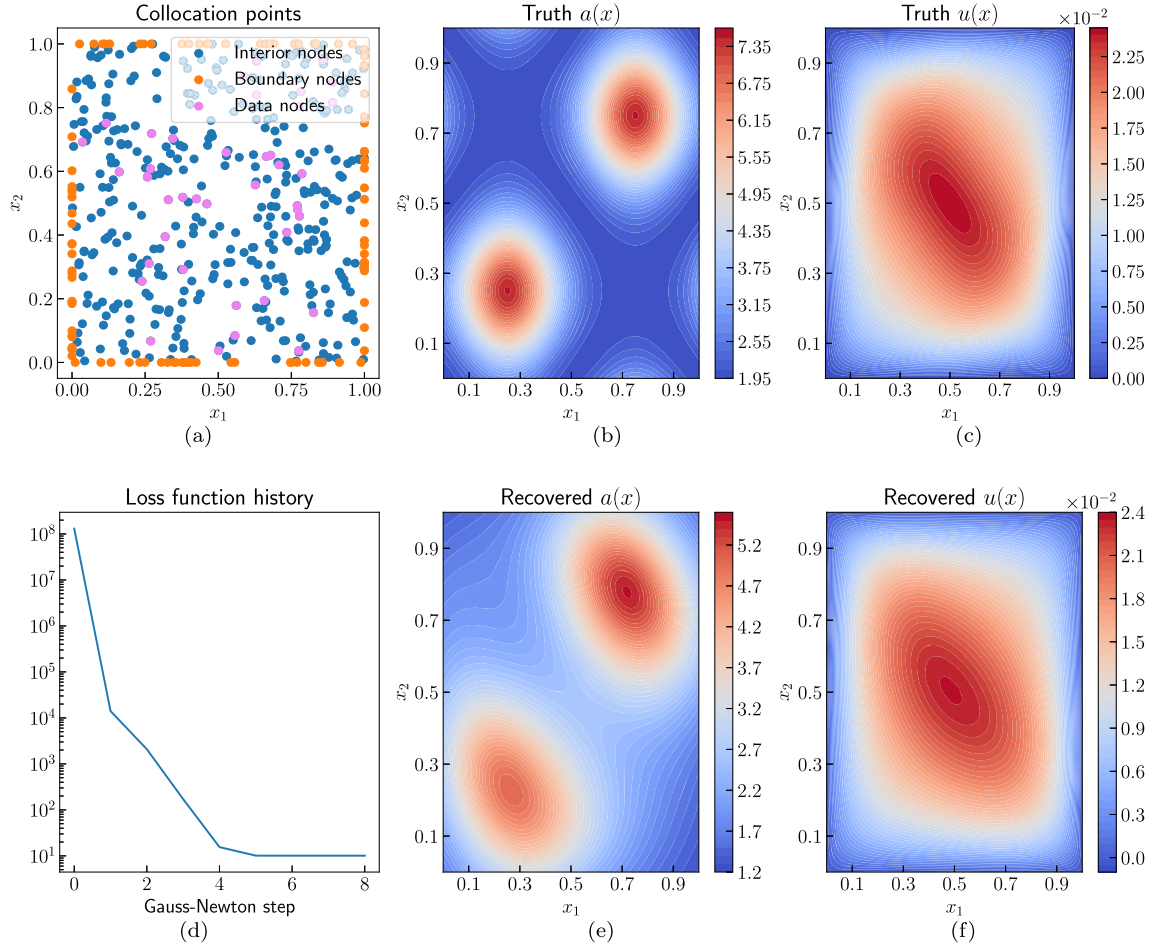


Fig. 5. Numerical results for the inverse Darcy flow: (a) an instance of uniformly sampled collocation points and data points; (d) Gauss–Newton iteration history; (b) true a ; (e) recovered a ; (c) true u ; (f) recovered u . Adaptive diagonal regularization (“nugget”) terms were added to the kernel matrix, with parameters $\eta = \bar{\eta} = 10^{-5}$ as outlined in Appendix A.5.

Generally, this condition is equivalent to the requirement that the nonlinear manifold $\mathcal{V} := \{v \in \mathcal{U} \mid F([\phi, v]) = F([\phi, u^*])\}$ lies on only one side of the tangent hyperplane supported by \mathbf{z}^\dagger with Θ^{-1} -normal direction \mathbf{z}^\dagger . Note that (5.1) is always satisfied with an equality if the PDE is linear, i.e., F is affine. Then the following proposition states that the minimizer u^\dagger of (3.2) is not only unique, it is also the minimax optimal approximation of u^* in \mathcal{V} in the sense of optimal recovery [40].

Proposition 5.1. *Under condition (5.1) it holds true that:*

1. u^\dagger is unique.
2. $\|v - u^\dagger\|^2 \leq \|v\|^2 - \|u^\dagger\|^2$ for any $v \in \mathcal{V}$.
3. u^\dagger is one of the minimizers of the minimax loss $\inf_{u \in \mathcal{V}} \sup_{v \in \mathcal{V}, v \neq 0} \frac{\|v - u\|}{\|v\|}$.

Proof. For any $v \in \mathcal{V}$, we denote by $\bar{v} = \sum_{n=1}^N [\phi_n, v] \chi_n$, the GP MAP solution after observing the values $[\phi_n, v]$ for $1 \leq n \leq N$. By definition, $\bar{v} \in \mathcal{V}$ and $[\phi, v - \bar{v}] = 0$. Thus, $v - \bar{v}$ is orthogonal to the linear span of χ_n in the $\|\cdot\|$ norm. It follows that $\|v - u^\dagger\|^2 = \|v - \bar{v}\|^2 + \|\bar{v} - u^\dagger\|^2$. Furthermore writing $\mathbf{z} = [\phi, v]$, we have

$$\|\bar{v} - u^\dagger\|^2 = \|\bar{v}\|^2 + \|u^\dagger\|^2 - 2\mathbf{z}^T \Theta^{-1} \mathbf{z}^\dagger = \|\bar{v}\|^2 - \|u^\dagger\|^2 - 2(\mathbf{z} - \mathbf{z}^\dagger)^T \Theta^{-1} \mathbf{z}^\dagger \leq \|\bar{v}\|^2 - \|u^\dagger\|^2,$$

where we used the relation $\|u^\dagger\|^2 = (\mathbf{z}^\dagger)^T \Theta^{-1} \mathbf{z}^\dagger$ and the assumed condition (5.1). Thus, we deduce that $\|v - u^\dagger\|^2 \leq \|v - \bar{v}\|^2 + \|\bar{v}\|^2 - \|u^\dagger\|^2 = \|v\|^2 - \|u^\dagger\|^2$ according to the orthogonality. This leads to the second point of the proposition, and the first point follows directly.

The proof of the third point is similar to that of [47, Thm. 18.2]. Here we shall prove that for the given minimax loss, the optimal value is 1 and the minimizer is $\sum_{n=1}^N z_n \chi_n$ for any \mathbf{z} satisfying $F(\mathbf{z}) = \mathbf{y}$. To achieve this, first observe that the

loss is always less or equal to 1 because for $v \in \mathcal{V}$, $\|v - \bar{v}\|^2 / \|v\|^2 \leq 1$ and $\bar{v} \in \mathcal{V}$. Thus, the optimal value is not larger than 1.

Moreover, let v^\perp be a nonzero element in \mathcal{U} such that $[\phi, v^\perp] = 0$. Consider $v_c = cv^\perp + \sum_{n=1}^N z_n \chi_n$ for any \mathbf{z} satisfying $F(\mathbf{z}) = \mathbf{y}$. Taking $u = \sum_{n=1}^N z_n \chi_n$, we have that

$$\frac{\|v_c - u\|^2}{\|v_c\|^2} = \frac{c^2 \|v^\perp\|^2}{c^2 \|v^\perp\|^2 + \|u\|^2};$$

this term converges towards 1 as $c \rightarrow \infty$. Thus, we get that the optimal value is exactly 1 and $u = \sum_{n=1}^N z_n \chi_n$ are minimizers. ■

Note that the above proposition implies the uniqueness of the minimizer u^\dagger but does not imply the uniqueness of the solution u^* . It is also interesting to observe that condition (5.1) only involves the nonlinear map F and not the differential operators defining the PDE.

5.2. Error estimates

Here we present rigorous error estimates in the PDE setting of Subsection 3 by bounding the error between the minimizer u^\dagger of (3.2) and the true solution u^* . The main idea behind our error bounds is to connect the conditional covariance of the underlying GP to the pointwise error of the conditional MAP estimator.

In the setting of Section 2.2, consider $\ker(\phi) := \{u \in \mathcal{U} | [\phi, u] = 0\}$. Then the conditional covariance operator \mathcal{K}_ϕ of Proposition 2.1 coincides with the short of the operator \mathcal{K} to $\ker(\phi)$ [2,46]: this is a symmetric and positive operator from \mathcal{U}^* to \mathcal{U} identified via

$$[\varphi, \mathcal{K}_\phi \varphi] = \inf_{\bar{\varphi} \in \text{span } \phi} \langle \varphi - \bar{\varphi}, \varphi - \bar{\varphi} \rangle_* = \inf_{\bar{\varphi} \in \text{span } \phi} [\varphi - \bar{\varphi}, \mathcal{K}(\varphi - \bar{\varphi})], \quad \forall \varphi \in \mathcal{U}^*, \quad (5.2)$$

where we used the shorthand notation $\text{span } \phi = \text{span}\{\phi_1, \dots, \phi_N\}$. This identification holds true even when the ϕ_n are not linearly independent. With the notion of the shorted operator at hand we can now present an abstract error bound on the action of dual elements on u^\dagger and u^* .

Proposition 5.2. *Let \mathcal{H} be a Banach space endowed with a quadratic norm such that the embedding $\mathcal{U} \subset \mathcal{H}$ is compact and such that $\mathcal{H}^* \subset \mathcal{U}^*$ where \mathcal{H}^* is defined using the same duality pairing as for \mathcal{U}^* . Let $u^\dagger = \sum_{n=1}^N z_n^\dagger \chi_n$ be a minimizer of (3.2) approximating the unique solution u^* of (3.1). Fix $\varphi \in \mathcal{H}^*$ and define ϕ as in (3.6). Then it holds true that*

$$|[\varphi, u^\dagger] - [\varphi, u^*]| \leq \sigma(\varphi) \|u^*\| + \epsilon \|\varphi\|_{\mathcal{H}^*}, \quad (5.3)$$

where $\sigma^2(\varphi) := \mathcal{K}(\varphi, \varphi) - \mathcal{K}(\varphi, \phi) \mathcal{K}(\phi, \phi)^{-1} \mathcal{K}(\phi, \varphi)$ and

$$\epsilon^2 := \begin{cases} \max_{\mathbf{z} \in \mathbb{R}^N} (\mathbf{z} - \mathbf{z}^\dagger)^T A (\mathbf{z} - \mathbf{z}^\dagger) \\ \text{s.t.} \quad F(\mathbf{z}) = \mathbf{y}, \\ \mathbf{z}^T \Theta^{-1} \mathbf{z} \leq \|u^*\|^2, \end{cases}$$

where A is the $N \times N$ positive symmetric matrix with entries $A_{i,j} = \langle \chi_i, \chi_j \rangle_{\mathcal{H}}$. Furthermore, it also holds true that

$$\|u^\dagger - u^*\| \leq \inf_{\phi \in \text{span } \phi} \|u^* - \mathcal{K}_\phi \phi\| + \epsilon. \quad (5.4)$$

Proof. Define $\mathbf{v} := [\phi, u^*]$ and $\bar{u} := \mathbf{v}^T \Theta^{-1} \mathcal{K}_\phi \phi \in \mathcal{U}$. By triangle inequality we have

$$|[\varphi, u^*] - [\varphi, u^\dagger]| \leq |[\varphi, u^*] - [\varphi, \bar{u}]| + |[\varphi, \bar{u}] - [\varphi, u^\dagger]|.$$

Since $[\phi, u^*] = [\phi, \bar{u}]$ then by [43, Thm. 5.1],

$$|[\varphi, u^*] - [\varphi, \bar{u}]| = \inf_{\bar{\varphi} \in \text{span } \phi} |[\varphi - \bar{\varphi}, u^* - \bar{u}]| \leq \inf_{\bar{\varphi} \in \text{span } \phi} \|\varphi - \bar{\varphi}\|_* \|u^* - \bar{u}\|.$$

Then (5.2) implies that $\inf_{\bar{\varphi} \in \text{span } \phi} \|\varphi - \bar{\varphi}\|_* = \sigma(\varphi)$. Since \bar{u} minimizes $\|u\|$ subject to $[\phi, u] = [\phi, u^*]$, we have $\|u^* - \bar{u}\|^2 = \|u^*\|^2 - \|\bar{u}\|^2$, which leads to $|[\varphi, u^*] - [\varphi, \bar{u}]| \leq \sigma(\varphi) \|u^*\|$. Furthermore, $|[\varphi, \bar{u}] - [\varphi, u^\dagger]| \leq \|\varphi\|_{\mathcal{H}^*} \|\bar{u} - u^\dagger\|_{\mathcal{H}}$ and observe that $\|\bar{u} - u^\dagger\|_{\mathcal{H}}^2 = (\mathbf{z} - \mathbf{z}^\dagger)^T A (\mathbf{z} - \mathbf{z}^\dagger)$. Since $\|\bar{u}\| \leq \|u^*\|$ and $\|u^\dagger\| \leq \|u^*\|$, and $F([\phi, \bar{u}]) = \mathbf{y}$ we deduce that $\|\bar{u} - u^\dagger\|_{\mathcal{H}}$ is bounded by the supremum of $\|v' - u^\dagger\|_{\mathcal{H}}$ over all $v' \in \mathcal{U}$ that belong to the ball of radius $\|u^*\|$ and satisfy the PDE constraints at the collocation points. Inequality (5.4) follows from $\|u^\dagger - u^*\| \leq \|u^* - \bar{u}\| + \epsilon$ and the fact that \bar{u} is the projection of u^\dagger onto $\text{span } \phi$ [47, Thm. 12.3]. ■

In the case where $\mathcal{H} \subset C^t(\Omega) \cap C^t(\overline{\Omega})$ we can take φ to be of the form $\delta_x \circ L$ where L is a differential operator of order at most t for $x \in \Omega$, and t' for $x \in \partial\Omega$. Then Proposition 5.2 allows us to bound the pointwise error of u^\dagger and its derivatives. As in Theorem 3.3, the compact embedding of \mathcal{U} into \mathcal{H} guarantees the convergence of $\sigma(\varphi)$ and ϵ towards zero as the collocation points become dense in Ω and $\partial\Omega$. Note that, by (5.4), the strong approximation error between u^\dagger and u^* is controlled by the sum between ϵ and the best approximation error in $\text{span } \mathcal{K}\phi$. Controlling this best approximation error is typically done in the dual space in the linear setting [47, Sec. 14.8]. In the nonlinear setting, it requires controlling the regularity of the solution (analyzing the underlying PDE). Moreover if the fill distance of the collocation points is h , and if the norm $\|\cdot\|$ satisfies the embedding inequality $\|\cdot\| \leq C\|\cdot\|_{H^s(\Omega)}$, then [47, Lem. 14.39] implies $\|u^* - \bar{u}\| \leq Ch^s \|u^*\|_{H^{2s}(\Omega)}$.

5.3. Learning the kernel

An important factor in the accuracy and performance of our method is the choice of the kernel K (and in turn, the operator \mathcal{K}). The importance of the kernel choice is readily visible in Theorem 3.3 through the assumption that u^* , the solution of the PDE, should belong to the \mathcal{U} the RKHS corresponding to K . As discussed in [43], even for linear PDEs, if the underlying kernel is not adapted to the solution space of the PDE, then the resulting method can perform arbitrarily badly [3]. This point stresses the importance of learning (or possibly programming [49,45]) an informed kernel beyond enforcing the PDE at collocation points. While the problem of selecting the kernel is well-understood if the solution is smooth, or the PDE is linear with possibly rough coefficients [43], it can be challenging when the underlying PDE is nonlinear, and the solution is irregular. Since our approach employs an explicit kernel it allows us to learn the kernel directly using kernel (hyperparameter) learning techniques such as maximum likelihood estimation (MLE) [10], MAP estimation [45], or cross-validation (CV) [21,50,10]. Adaptation of these techniques for solution of nonlinear PDEs is an attractive direction for future research.

5.4. Non-Gaussian priors

Although we have employed a Gaussian prior in the design of our nonlinear solver and in our Bayesian IPs, our methodology can naturally be generalized to priors that can be obtained as nonlinear transformations of Gaussian measures (e.g., instead of replacing the solution u^* of a nonlinear PDE by a GP ξ , one could replace it by $G(\xi)$ where G is a nonlinear function with the appropriate degree of regularity so that the pointwise derivatives of $G(\xi)$ remain well-defined). This approach can potentially lead to efficient methods for computing MAP estimators of Bayesian IPs with certain non-Gaussian priors, a contemporary problem in the uncertainty quantification and IP literatures.

CRedit authorship contribution statement

Yifan Chen: Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Bamdad Hosseini:** Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Houman Owhadi:** Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Supervision, Writing – original draft, Writing – review & editing. **Andrew M. Stuart:** Formal analysis, Methodology, Writing – review & editing.

Declaration of competing interest

George Karniadakis, Houman Owhadi and Andrew M Stuart are CoPIs on a MURI grant (number 19) awarded by AFOSR. We do not think that this constitutes a conflict of interest, but we are disclosing this for the sake of full transparency and to prevent ambiguity.

Acknowledgements

The authors gratefully acknowledge support by the Air Force Office of Scientific Research under MURI award number FA9550-20-1-0358 (Machine Learning and Physics-Based Modeling and Simulation). HO also acknowledges support by the Air Force Office of Scientific Research under award number FA9550-18-1-0271 (Games for Computation and Learning). YC is also grateful for support from the Caltech Kortchak Scholar Program.

Appendix A. Diagonal regularization of kernel matrices

In the context of GP regression it is common to regularize the kernel matrices by addition of a small diagonal term; in that context, doing so has the interpretation of assuming small Gaussian noise on top of the observations. This diagonal regularization is sometimes referred to as a “nugget”. Here we discuss a related approach to regularizing kernel matrices (Θ and $\tilde{\Theta}$) by perturbing them slightly; for brevity we use the terminology “nugget” throughout. In Appendix A.1 we present an adaptive approach to constructing a family of nugget terms that is tailored to our kernel matrices. Appendices A.2 through A.4 gather the detailed choices of nugget terms for the experiments in Subsections 3.5.1 through 3.5.3. Appendix A.5 contains the same details for the experiments in Subsection 4.4.

Table 4

Comparison of solution errors between standard nugget terms and our adaptive nugget terms for the nonlinear elliptic PDE (1.1). Collocation points are uniformly sampled with $M = 1024$ and $M_\Omega = 900$ with a Gaussian kernel with lengthscale parameter $\sigma = 0.2$. Results are averaged over 10 realizations of the random collocation points. The maximum Gauss-Newton iteration was 5.

η	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
$\Theta + \eta I$: L^2 error	7.77e-02	4.46e-02	2.65e-02	1.56e-02	1.32e-02	1.46e-02
$\Theta + \eta I$: L^∞ error	6.43e-01	3.13e-01	1.99e-01	1.47e-01	1.33e-01	1.43e-01
$\Theta + \eta R$: L^2 error	8.49e-02	9.29e-03	9.10e-03	3.34e-03	1.01e-03	3.36e-04
$\Theta + \eta R$: L^∞ error	4.02e-01	7.86e-02	5.58e-02	2.21e-02	7.17e-03	3.87e-03
η	10^{-7}	10^{-8}	10^{-9}	10^{-10}	10^{-11}	10^{-12}
$\Theta + \eta I$: L^2 error	1.37e-02	8.623e-03	1.01e-02	1.92e-02	nan	nan
$\Theta + \eta I$: L^∞ error	1.81e-01	8.28e-02	1.07e-01	3.05e-01	nan	nan
$\Theta + \eta R$: L^2 error	1.55e-04	7.05e-05	4.56e-05	6.30e-06	1.73e-06	8.31e-07
$\Theta + \eta R$: L^∞ error	2.41e-03	1.07e-03	7.66e-04	8.92e-05	2.62e-05	1.17e-05

A.1. An adaptive nugget term

One of the main computational issues in implementing our methodology is that the kernel matrix Θ is ill-conditioned. As a consequence, we need to regularize this matrix to improve the stability of these algorithms. This regularization may introduce an accuracy floor, so it is important to choose the regularization term so that it has a small effect on accuracy – there is thus an accuracy-stability tradeoff. A traditional strategy for achieving this goal is to add a nugget term ηI to Θ , where $\eta > 0$ is a small number, and I is the identity matrix. By choosing a suitable η , the condition number of $\Theta + \eta I$ can be improved significantly. However, there is an additional level of difficulty in our method since the matrix Θ contains multiple blocks whose spectral properties can differ by orders of magnitude, since they can involve different orders of derivatives of the kernel function K . This observation implies that adding ηI , which is uniform across all blocks, may be suboptimal in terms of the accuracy-stability tradeoff.

In what follows we adopt the same notation as Subsection 3.4.1. To address the ill-conditioning of Θ , we consider adding an adaptive block diagonal nugget term. More precisely, without loss of generality we can assume $\Theta^{(1,1)}$ corresponds to the pointwise measurements, i.e., $L_1^X = \delta_x$, then, we construct a block diagonal matrix

$$R = \begin{bmatrix} I & & & \\ & \frac{\text{tr}(\Theta^{(2,2)})}{\text{tr}(\Theta^{(1,1)})} I & & \\ & & \ddots & \\ & & & \frac{\text{tr}(\Theta^{(Q,Q)})}{\text{tr}(\Theta^{(1,1)})} I \end{bmatrix},$$

where we reweight the identity matrix in each diagonal block by a factor of the trace ratio between $\Theta^{(q,q)}$ and $\Theta^{(1,1)}$. With this matrix, the adaptive nugget term is defined as ηR with a global nugget parameter $\eta > 0$. We find that once the parameter η is chosen suitably, then our Gauss–Newton algorithm converges quickly and in a stable manner. During computations, we can compute the Cholesky factors of $\Theta + \eta R$ offline and use back-substitution to invert them in each iteration of Gauss–Newton.

Example NE. For the numerical experiments in Subsection 1.1.4 pertaining to the nonlinear elliptic PDE (1.1), we observed that Θ has two distinct diagonal blocks, i.e., one block corresponding to the pointwise evaluation functions and with entries $K(\mathbf{x}_m, \mathbf{x}_i)$ and another block corresponding to pointwise evaluations of the Laplacian operator and with entries $\Delta^X \Delta^{X'} K(\mathbf{x}, \mathbf{x}')|_{(\mathbf{x}_m, \mathbf{x}_i)}$. With $M = 1024$ collocation points, the trace ratio between these blocks was of order 4000. Thus, the difference between I and R is significant. Our experiments also showed that if we only add ηI to regularize the matrix, then choosing η as large as $O(10^{-4})$ was needed to get meaningful results, while using the nugget term ηR , we could choose $\eta = 10^{-9}$ which leads to significantly improved results. We further explore these details below and in particular in Table 4. \diamond

A.2. Choice of nugget terms for the nonlinear elliptic PDE

Below we discuss the choice of the nugget term in the numerical experiments of Subsection 3.5.1. The results in Fig. 2 and Table 1 were obtained by employing the adaptive nugget term of Appendix A.1 with global parameters $\eta = 10^{-13}$ and $\eta = 10^{-12}$ respectively.

We also compared our adaptive nugget term to more standard choices, i.e., nugget terms of the form ηI against our adaptive nugget term ηR with the nonlinearity $\tau(u) = u^3$. Cholesky factorization was applied to the regularized matrix and the subsequent Gauss–Newton iterations were employed to obtain the solutions. The L^2 and L^∞ errors of the converged solutions are shown in Table 4. The results were averaged over 10 instances of a random sampling of the collocation

points. Here, “nan” means the algorithm was unstable and diverged. To obtain these results we terminated all Gauss-Newton iterations after 5 steps. Due to randomness in the initial guess, and in examples where random collocation points were used due to this too, we observed that some random trials did not converge in 5 steps. This variation also explains the non-monotonic behavior of the error in Table 4 as η decreases. These effects were more profound for the standard nugget term. Besides these small variations our results clearly demonstrate the superior accuracy and stability that is provided by our adaptive nugget term versus the standard nugget choice.

A.3. Choice of nugget terms for Burger's equation

For the numerical experiments in Subsection 3.5.2 we primarily used our adaptive nugget term as outlined in Appendix A.1. For the results in Fig. 3 we used a global nugget parameter $\eta = 10^{-10}$. For the convergence analysis in Table 2 we varied η for different number of collocation points to achieve better performance. More precisely, for $M \leq 1200$ we used a larger nugget $\eta = 10^{-5}$ and for $M \geq 2400$ we used $\eta = 10^{-10}$. Choosing a smaller nugget for small values of M would still yield equally accurate results but required more iterations of the Gauss-Newton algorithm.

A.4. Choice of nugget terms for the Eikonal equation

Our numerical experiments in Subsection 3.5.3 also used the adaptive nugget of Appendix A.1. Indeed, we followed a similar approach to choosing the global parameter η as in the case of Burger's equation outlined in Appendix A.3 above.

For the results in Fig. 4 we used $\eta = 10^{-10}$. For the convergence analysis in Table 3 we varied η for different values of M , i.e., we chose $\eta = 10^{-5}$ for $M \leq 1200$ and $\eta = 10^{-10}$ for $M \geq 2400$. Analogously to the Burger's experiment we observed that smaller values of η for smaller choices of M cause the Gauss-Newton iterations to converge more slowly. Thus varying η with M improved the efficiency of our framework.

A.5. Choice of nugget terms for Darcy flow

Both of the matrices $\Theta, \tilde{\Theta}$ outlined in Subsection 4.1 are dense and ill-conditioned in the IP setting and so an appropriate nugget term should be chosen to regularize them. In the IP setting we propose to add adaptive nuggets for both $\Theta, \tilde{\Theta}$ using the same strategy as in Appendix A.1, except that the nuggets are constructed independently for each matrix. To this end we set $\Theta \leftarrow \Theta + \eta R$ and $\tilde{\Theta} \leftarrow \tilde{\Theta} + \tilde{\eta} \tilde{R}$, where the $\tilde{\eta}, \tilde{R}$ denote the global nugget parameter and the re-weighted identity matrix corresponding to $\tilde{\Theta}$. For the numerical experiments in Fig. 5 we used $\eta = \tilde{\eta} = 10^{-5}$.

References

- [1] R.A. Adams, J.J. Fournier, Sobolev Spaces, Elsevier, 2003.
- [2] W. Anderson Jr, G. Trapp, Shorted operators. II, SIAM J. Appl. Math. 28 (1975) 60–71.
- [3] I. Babuška, J. Osborn, Can a finite element method perform arbitrarily badly?, Math. Comput. 69 (2000) 443–462.
- [4] Y. Bar-Sinai, S. Hoyer, J. Hickey, M.P. Brenner, Learning data-driven discretizations for partial differential equations, Proc. Natl. Acad. Sci. 116 (2019) 15344–15349.
- [5] K. Bhattacharya, B. Hosseini, N.B. Kovachki, A.M. Stuart, Model reduction and neural networks for parametric PDEs, SMAI J. Comput. Math. 7 (2021) 121–157.
- [6] V.I. Bogachev, Gaussian Measures, American Mathematical Society, 1998.
- [7] G. Boncoraglio, C. Farhat, Active manifold and model reduction for multidisciplinary analysis and optimization, in: AIAA Scitech 2021 Forum, 2021, p. 1694.
- [8] H. Brezis, Functional Analysis, Sobolev Spaces and Partial Differential Equations, Springer Science & Business Media, 2010.
- [9] B. Calderhead, M. Girolami, N.D. Lawrence, Accelerating Bayesian inference over nonlinear differential equations with Gaussian processes, in: Advances in Neural Information Processing Systems, 2009, pp. 217–224.
- [10] Y. Chen, H. Owthadi, A. Stuart, Consistency of empirical Bayes and kernel flow for hierarchical parameter estimation, Math. Comput. (2021).
- [11] A. Cochoco, R. Unbehauen, Neural Networks for Optimization and Signal Processing, John Wiley & Sons, Inc., 1993.
- [12] J. Cockayne, C. Oates, T. Sullivan, M. Girolami, Probabilistic numerical methods for PDE-constrained Bayesian inverse problems, AIP Conf. Proc. 1853 (2017) 060001.
- [13] J. Cockayne, C.J. Oates, T.J. Sullivan, M. Girolami, Bayesian probabilistic numerical methods, SIAM Rev. 61 (2019) 756–789.
- [14] S.L. Cotter, M. Dashti, A.M. Stuart, Approximation of Bayesian inverse problems for PDEs, SIAM J. Numer. Anal. 48 (2010) 322–345.
- [15] S.L. Cotter, G.O. Roberts, A.M. Stuart, D. White, MCMC methods for functions: modifying old algorithms to make them faster, Stat. Sci. (2013) 424–446.
- [16] M. Dashti, K.J. Law, A.M. Stuart, J. Voss, MAP estimators and their consistency in Bayesian nonparametric inverse problems, Inverse Probl. 29 (2013) 095017.
- [17] M. Dashti, A.M. Stuart, The Bayesian approach to inverse problems, in: R. Ghanem, D. Higdon, H. Owthadi (Eds.), Handbook of Uncertainty Quantification, Springer International Publishing, 2016, pp. 1–118.
- [18] P. Diaconis, Bayesian numerical analysis, in: S.S. Gupta, J. Berger (Eds.), Statistical Decision Theory and Related Topics IV, Springer, 1988.
- [19] B. Djeridane, J. Lygeros, Neural approximation of PDE solutions: an application to reachability computations, in: Proceedings of the 45th IEEE Conference on Decision and Control, IEEE, 2006, pp. 3034–3039.
- [20] H.W. Engl, M. Hanke, A. Neubauer, Regularization of Inverse Problems, vol. 375, Springer Science & Business Media, 1996.
- [21] J. Friedman, T. Hastie, R. Tibshirani, The Elements of Statistical Learning, Springer, 2001.
- [22] I. Goodfellow, Y. Bengio, A. Courville, Y. Bengio, Deep Learning, vol. 1, MIT Press, 2016.
- [23] J. Han, A. Jentzen, E. Weinan, Solving high-dimensional partial differential equations using deep learning, Proc. Natl. Acad. Sci. 115 (2018) 8505–8510.
- [24] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

- [25] P. Hennig, M.A. Osborne, M. Girolami, Probabilistic numerics and uncertainty in computations, *Proc. R. Soc. A* 471 (2015) 20150142.
- [26] M. Hinze, R. Pinnau, M. Ulbrich, S. Ulbrich, *Optimization with PDE Constraints*, vol. 23, Springer Science & Business Media, 2008.
- [27] A. Jacot, F. Gabriel, C. Hongler, Neural tangent kernel: convergence and generalization in neural networks, in: *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, 2018, pp. 8580–8589.
- [28] J. Kaipio, E. Somersalo, *Statistical and Computational Inverse Problems*, vol. 160, Springer Science & Business Media, 2006.
- [29] G. Karniadakis, I. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, L. Yang, Physics-informed machine learning, *Nat. Rev. Phys.* (2021).
- [30] G. Kimeldorf, G. Wahba, Some results on Tchebycheffian spline functions, *J. Math. Anal. Appl.* 33 (1971) 82–95.
- [31] G.S. Kimeldorf, G. Wahba, A correspondence between Bayesian estimation on stochastic processes and smoothing by splines, *Ann. Math. Stat.* 41 (1970) 495–502.
- [32] K. Krischer, R. Rico-Martínez, I. Kevrekidis, H. Rotermund, G. Ertl, J. Hudson, Model identification of a spatiotemporally varying catalytic reaction, *AIChE J.* 39 (1993) 89–98.
- [33] I.E. Lagaris, A.C. Likas, D.I. Fotiadis, Artificial neural networks for solving ordinary and partial differential equations, *IEEE Trans. Neural Netw.* 9 (1998) 987–1000.
- [34] I.E. Lagaris, A.C. Likas, D.G. Papageorgiou, Neural-network methods for boundary value problems with irregular boundaries, *IEEE Trans. Neural Netw.* 11 (2000) 1041–1049.
- [35] F.M. Larkin, Gaussian measure in Hilbert space and applications in numerical analysis, *J. Math.* 2 (1972).
- [36] Z. Li, N. Kovachki, K. Azizzadenesheli, K. Bhattacharya, A. Stuart, A. Anandkumar, Fourier neural operator for parametric partial differential equations, in: *International Conference on Learning Representations*, 2020.
- [37] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, A. Stuart, K. Bhattacharya, A. Anandkumar, Multipole graph neural operator for parametric partial differential equations, *Adv. Neural Inf. Process. Syst.* (2020).
- [38] H. Liang, H. Wu, Parameter estimation for differential equation models using a framework of measurement error in regression models, *J. Am. Stat. Assoc.* 103 (2008) 1570–1583.
- [39] Z. Long, Y. Lu, X. Ma, B. Dong, PDE-net: learning PDEs from data, in: *International Conference on Machine Learning*, PMLR, 2018, pp. 3208–3216.
- [40] C.A. Micchelli, T.J. Rivlin, A survey of optimal recovery, in: C. Micchelli, T.J. Rivlin (Eds.), *Optimal Estimation in Approximation Theory*, Springer, 1977, pp. 1–54.
- [41] N.H. Nelsen, A.M. Stuart, The random feature model for input-output maps between Banach spaces, *arXiv preprint*, arXiv:2005.10224, 2020.
- [42] J. Nocedal, S. Wright, *Numerical Optimization*, Springer Science & Business Media, 2006.
- [43] H. Owghadi, Bayesian numerical homogenization, *Multiscale Model. Simul.* 13 (2015) 812–828.
- [44] H. Owghadi, Multigrid with rough coefficients and multiresolution operator decomposition from hierarchical information games, *SIAM Rev.* 59 (2017) 99–149.
- [45] H. Owghadi, Do ideas have shape? Plato's theory of forms as the continuous limit of artificial neural networks, *arXiv preprint*, arXiv:2008.03920, 2020.
- [46] H. Owghadi, C. Scovel, Conditioning Gaussian measure on Hilbert space, *J. Math. Statist. Anal.* 1 (2018).
- [47] H. Owghadi, C. Scovel, Operator-Adapted Wavelets, Fast Solvers, and Numerical Homogenization: From a Game Theoretic Approach to Numerical Approximation and Algorithm Design, Cambridge University Press, 2019.
- [48] H. Owghadi, C. Scovel, F. Schäfer, Statistical numerical approximation, *Not. AMS* 66 (10) (2019) 1608–1617.
- [49] H. Owghadi, C. Scovel, G.R. Yoo, Kernel mode decomposition and programmable/interpretable regression networks, *arXiv preprint*, arXiv:1907.08592, 2019.
- [50] H. Owghadi, G.R. Yoo, Kernel flows: from learning kernels from data into the abyss, *J. Comput. Phys.* 389 (2019) 22–47.
- [51] H. Owghadi, L. Zhang, Gamblets for opening the complexity-bottleneck of implicit schemes for hyperbolic and parabolic ODEs/PDEs with rough coefficients, *J. Comput. Phys.* 347 (2017) 99–128.
- [52] I. Palasti, A. Renyi, On interpolation theory and the theory of games, *MTA Mat. Kat. Int. Kozl.* 1 (1956) 529–540.
- [53] O. Perrin, P. Monestiez, Modelling of non-stationary spatial structure using parametric radial basis deformations, in: *GeoENV II—Geostatistics for Environmental Applications*, Springer, 1999, pp. 175–186.
- [54] H. Poincaré, *Calcul des Probabilités*, Georges Carrés, 1896.
- [55] J. Quinero-Candela, C.E. Rasmussen, A unifying view of sparse approximate Gaussian process regression, *J. Mach. Learn. Res.* 6 (2005) 1939–1959.
- [56] V.D. Rădulescu, *Qualitative Analysis of Nonlinear Elliptic Partial Differential Equations: Monotonicity, Analytic, and Variational Methods*, Hindawi Publishing Corporation, 2008.
- [57] M. Raissi, P. Perdikaris, G.E. Karniadakis, Numerical Gaussian processes for time-dependent and nonlinear partial differential equations, *SIAM J. Sci. Comput.* 40 (2018) A172–A198.
- [58] M. Raissi, P. Perdikaris, G.E. Karniadakis, Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, *J. Comput. Phys.* 378 (2019) 686–707.
- [59] M. Raissi, A. Yazdani, G.E. Karniadakis, Hidden fluid mechanics: learning velocity and pressure fields from flow visualizations, *Science* 367 (2020) 1026–1030.
- [60] J.O. Ramsay, G. Hooker, D. Campbell, J. Cao, Parameter estimation for differential equations: a generalized smoothing approach, *J. R. Stat. Soc., Ser. B, Stat. Methodol.* 69 (2007) 741–796.
- [61] R. Rico-Martínez, I.G. Kevrekidis, Continuous time modeling of nonlinear systems: a neural network-based approach, in: *IEEE International Conference on Neural Networks*, IEEE, 1993, pp. 1522–1525.
- [62] P.D. Sampson, P. Guttorp, Nonparametric estimation of nonstationary spatial covariance structure, *J. Am. Stat. Assoc.* 87 (1992) 108–119.
- [63] A. Sard, *Linear Approximation*, American Mathematical Society, 1963.
- [64] R. Schaback, H. Wendland, Kernel techniques: from machine learning to meshless methods, *Acta Numer.* 15 (2006) 543.
- [65] F. Schäfer, M. Katzfuss, H. Owghadi, Sparse Cholesky factorization by Kullback–Leibler minimization, *SIAM J. Sci. Comput.* 43 (2021) A2019–A2046.
- [66] F. Schäfer, T. Sullivan, H. Owghadi, Compression, inversion, and approximate PCA of dense kernel matrices at near-linear computational complexity, *Multiscale Model. Simul.* 19 (2021) 688–730.
- [67] M. Schober, D. Duvenaud, P. Hennig, Probabilistic ODE solvers with Runge-Kutta means, in: *28th Annual Conference on Neural Information Processing Systems*, 2014, pp. 739–747.
- [68] B. Schölkopf, A.J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press, 2018.
- [69] Y. Shin, J. Darbon, G.E. Karniadakis, On the convergence and generalization of physics informed neural networks, *arXiv preprint*, arXiv:2004.01806, 2020.
- [70] J. Sirignano, K. Spiliopoulos, DGM: a deep learning algorithm for solving partial differential equations, *J. Comput. Phys.* 375 (2018) 1339–1364.
- [71] J. Skilling, Bayesian solution of ordinary differential equations, in: C.R. Smith, G.J. Erickson, P.O. Neudorfer (Eds.), *Maximum Entropy and Bayesian Methods*, Springer, 1992, pp. 23–37.
- [72] J. Smoller, *Shock Waves and Reaction–Diffusion Equations*, Springer Science & Business Media, 2012.
- [73] M.L. Stein, The screening effect in kriging, *Ann. Stat.* 30 (2002) 298–323.
- [74] A.M. Stuart, Inverse problems: a Bayesian perspective, *Acta Numer.* 19 (2010) 451–559.

- [75] A.V. Sul'din, Wiener measure and its applications to approximation methods. I, *Izv. Vysš. Učebn. Zaved., Mat.* (1959) 145–158.
- [76] L.P. Swiler, M. Gulian, A.L. Frankel, C. Safta, J.D. Jakeman, A survey of constrained Gaussian process regression: approaches and implementation challenges, *J. Mach. Learn. Model. Comput.* 1 (2020).
- [77] E. Tadmor, A review of numerical methods for nonlinear partial differential equations, *Bull. Am. Math. Soc.* 49 (2012) 507–554.
- [78] J.F. Traub, G. Wasilkowski, H. Woźniakowski, E. Novak, Information-based complexity, *SIAM Rev.* 36 (1994) 514.
- [79] J.F. Traub, G.W. Wasilkowski, H. Woźniakowski, Average case optimality for linear problems, *Theor. Comput. Sci.* 29 (1984) 1–25.
- [80] T. Uchiyama, N. Sonehara, Solving inverse problems in nonlinear PDEs by recurrent neural networks, in: *IEEE International Conference on Neural Networks*, 1993, pp. 99–102.
- [81] R. van der Meer, C. Oosterlee, A. Borovykh, Optimally weighted loss functions for solving PDEs with neural networks, *arXiv preprint*, arXiv:2002.06269, 2020.
- [82] A.W. van der Vaart, J.H. van Zanten, Reproducing kernel Hilbert spaces of Gaussian priors, in: *Pushing the Limits of Contemporary Statistics: Contributions in Honor of Jayanta K. Ghosh*, Institute of Mathematical Statistics, 2008, pp. 200–222.
- [83] A.V. Vecchia, Estimation and model identification for continuous spatial processes, *J. R. Stat. Soc. B* 50 (1988) 297–312.
- [84] G. Wahba, *Spline Models for Observational Data*, SIAM, 1990.
- [85] S. Wang, H. Wang, P. Perdikaris, On the eigenvector bias of Fourier feature networks: from regression to solving multi-scale PDEs with physics-informed neural networks, *Comput. Methods Appl. Mech. Eng.* 384 (2021) 113938.
- [86] S. Wang, X. Yu, P. Perdikaris, When and why PINNs fail to train: a neural tangent kernel perspective, *arXiv preprint*, arXiv:2007.14527, 2020.
- [87] E. Weinan, B. Yu, The deep Ritz method: a deep learning-based numerical algorithm for solving variational problems, *Commun. Math. Stat.* 6 (2018) 1–12.
- [88] H. Wendland, *Scattered Data Approximation*, Cambridge University Press, 2004.
- [89] C.K.I. Williams, C.E. Rasmussen, *Gaussian Processes for Machine Learning*, The MIT Press, 2006.
- [90] A.G. Wilson, Z. Hu, R. Salakhutdinov, E.P. Xing, Deep kernel learning, in: *Artificial Intelligence and Statistics*, in: PMLR, vol. 51, 2016, pp. 370–378.
- [91] X. Zhang, K.Z. Song, M.W. Lu, X. Liu, Meshless methods based on collocation with radial basis functions, *Comput. Mech.* 26 (2000) 333–343.
- [92] Y. Zhu, N. Zabaras, Bayesian deep convolutional encoder–decoder networks for surrogate modeling and uncertainty quantification, *J. Comput. Phys.* 366 (2018) 415–447.