GAUSSIAN PROCESS PRIORS FOR BOUNDARY VALUE PROBLEMS OF LINEAR PARTIAL DIFFERENTIAL EQUATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose Boundary Ehrenpreis–Palamodov Gaussian Processes (B-EPGPs), a probabilistic framework for constructing Gaussian Processes (GP) priors for linear constant-coefficient Partial Differential Equations (PDE) with linear boundary conditions that can be conditioned on a finite data set. Starting from the Ehrenpreis–Palamodov representation, we learn the free parameters from data and enforce boundary conditions analytically for piecewise-flat boundaries. This yields priors and posteriors whose sample paths strictly satisfy both PDE and boundary conditions by construction. We provide constructive examples, formal correctness proofs, and experiments showing improved accuracy and reduced runtime/memory compared to existing PDE baselines.

1 INTRODUCTION

The efficient and accurate solution of linear partial differential equations (PDEs) with constant coefficients remains a computationally and theoretically critical endeavor. These canonical systems encompass foundational models like the heat, wave, Laplace, or Poisson equations and govern a vast array of processes across engineering, physics, and finance. Because real-world applications frequently involve high-dimensional state spaces or highly irregular spatial domains where traditional numerical methods (e.g. finite element or finite difference methods) suffer from severe computational bottlenecks such as the curse of dimensionality, the development of fast, scalable, and generalizable ML-based solvers for linear PDEs remains as highly relevant today as ever.

Classically, systems of PDEs have been *solved* using numerical methods, which require the solution to be uniquely determined. This uniqueness is typically ensured by prescribing a sufficient number of initial or boundary conditions. Neural operators adopt a similar strategy, often learning the evolution of PDE solutions by mapping the system state at time t to the state at time $t+1$, given appropriate initial conditions. While such methods significantly reduce computational inference cost, their accuracy typically falls short of that achieved by traditional numerical solvers.

A second line of research in machine learning abandons the requirement of uniqueness and instead seeks a "good" approximate solution that fits observed data, i.e. *regression* inside of the solution set of PDEs. This paper falls in this line of work. A prominent example is physics-informed neural networks (PINNs) (Raissi et al., 2019), which combine standard regression loss with an additional penalty for violating the PDE at sampled points in the domain. Many Gaussian process (GP) models for linear PDEs also follow this approach.

GPs (Rasmussen & Williams, 2006) are a standard choice for functional priors, offering robust regression with few data points and calibrated uncertainty estimates. Their covariance structure can encode linear properties, such as derivatives (Swain et al., 2016), enabling the construction of GPs whose realizations lie in the solution set of linear PDEs with constant coefficients (Lange-Hegermann, 2018; Jidling et al., 2017; Macêdo & Castro, 2008; Scheuerer & Schlather, 2012; Wahlström et al., 2013; Solin et al., 2018; Jidling et al., 2018; Särkkä, 2011), provided the system is controllable, i.e., admits potentials. This construction was extended beyond controllable systems (Harkonen et al., 2023). These methods yield machine learning models whose predictions are physics-*constrained* rather than physics-*informed*: all regression functions are exact solutions of

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

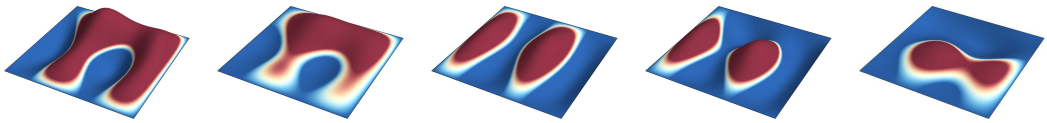


Figure 1: Dynamics of the 2D wave equation $\partial_t^2 u - \partial_x^2 u - \partial_y^2 u = 0$ with boundary condition $u = 0$ on a rectangular spatial domain at times $t = 0, 1, 2, 3, 4$. This figure shows a sample drawn from our B-EPGP construction. Every sample analytically satisfies both the differential equation and the boundary conditions, and can be conditioned on any finite set of observations.

the PDE system. As a result, they surpass PINNs in accuracy by several orders of magnitude (Harkonen et al., 2023). In practice, PDE systems are typically accompanied by both data and (initial or) boundary conditions, which we survey in Section 3.

This paper builds GP priors, which we call Boundary Ehrenpreis–Palamodov Gaussian Processes (**B-EPGP**s), by restricting the Ehrenpreis–Palamodov representation (Figure 1). The realizations of these GPs satisfy both a linear constant-coefficient PDE and homogeneous linear boundary conditions *exactly*. Hence, its inductive bias makes it data-efficient for regression.

Our main contributions are:

1. We provide explicit constructions of such priors for several representative PDEs, covering Dirichlet, Neumann, and Robin conditions on half-spaces and polyhedral domains.
2. We show that B-EPGP matches or exceeds classical solvers in accuracy while satisfying the PDE exactly, and remains accurate with as few as 10 data points.
3. Empirically, we show that B-EPGP outperforms GP-baselines and neural operators, and, unlike EPGP, scales to higher dimensions where EPGP exhausts GPU memory.
4. We prove that, for half-space Dirichlet/Neumann problems, these GP priors have realizations that are dense in the PDE solution space; we conjecture this holds more generally.

We outline the B-EPGP method in Section 4. Reconstructions from data are provided in various domains with polyhedral boundary in Sections 4.2, 5, 6, A. Comparisons to state-of-the-art methods are in Section 5. Theoretical guarantees are provided in Theorem 1 proved in Appendix C.4.

2 The Problem

We construct probability measures on the space of solutions of differential systems

$$\begin{cases} A(\partial)u = 0 & \text{in } \mathbb{R}^{n-1} \times (0, \infty) \\ B(\partial)u = 0 & \text{on } \mathbb{R}^{n-1} \times \{0\}, \end{cases} \tag{BVP}$$

where $A, B \in \mathbb{C}[\partial]$ are linear differential operators with complex constant coefficients for $\partial = (\partial_1, \partial_2, \ldots, \partial_n)$ the vector of all real partial derivatives in n variables, and $u \colon \mathbb{R}^n \to \mathbb{C}$ is the unknown solution of the system. The first equation in BVP encodes a homogeneous partial differential equation in the halfspace $\{x \in \mathbb{R}^n \colon x_n > 0\}$, while the second equation encodes a boundary condition on the hyperplane $\{x \in \mathbb{R}^n \colon x_n = 0\}$. For example A can be the Laplace, heat, and wave operators $\partial_1^2 + \partial_2^2 + \partial_3^2$, $\partial_1 - \partial_2^2 - \partial_3^2$, and $\partial_1^2 - \partial_2^2 - \partial_3^2$ respectively ($n = 3$) and B can represent the Dirichlet and Neumann boundary conditions $B(\partial)u = u$ and $B(\partial)u = \partial_3 u$ respectively.

The system BVP covers numerous differential constraints arising in scientific continuum models, from magnetism, to fluid flow, seismology, finance, and many others (Monk, 2003; Becker et al., 1999; Shearer, 2019; Black & Scholes, 1973; Bellini et al., 1979). Typically, the solution space of BVP is infinite dimensional, which makes the definition of probability measures on it a highly nontrivial problem (Bogachev, 1998). Having access to this probability measure helps (i) generate and simulate solutions to the system BVP; (ii) approximate any solution to the system; (iii) obtain

data-driven predictions using regression on sparse data; (iv) construct numerical solvers for well-posed problems if data is more abundant.

Importantly, our method shows how we can incorporate both *hard constraints* in BVP directly on the probability measure we construct. This is perfectly compatible with incorporating other soft or hard constraints such as $C(\partial)u = f$, but is not designed to construct approximate solutions of such inhomogeneous equations. The geometry of the halfspace is restrictive, but it is the fundamental starting point to understanding how boundary constraints influence a system or model. It is typical in the development of boundary value problems to study halfspaces first, then polyhedra—which we give example implementation in Sections 5.2 and A.1— and then approximate curved domains with polyhedra (Agmon et al., 1959; Ciarlet, 2002; Grisvard, 2011).

3 Related ML Approaches to PDEs with boundary conditions

Scientific Machine Learning, the intersection of machine learning and PDEs, has grown into a vibrant research field with a variety of modeling paradigms. This section surveys classes of methods, including neural operators, GPs, and hybrid models, placing our B-EPGP framework in context.

A dominant class of approaches uses neural networks either to approximate the solution directly or as implicit function approximators. A foundational work in this domain is that of physics-informed neural networks (PINNs) (Raissi et al., 2019), which penalize deviation from the PDE operator in the loss function. Variants and improvements of this idea abound, such as variational PINNs (Kharazmi et al., 2021), Fourier PINNs (Wang et al., 2021), and gradient-enhanced PINNs (Yu et al., 2022). These approaches train neural networks to minimize the PDE and approximate data at the same time. For a similar approach to GPs, see (Chen et al., 2022).

PINNs with hard constraints—often called *Ansatz PINNs*—enforce boundary and initial conditions exactly by constructing a trial solution $\hat{u} = g + h\,N_\theta$, where g satisfies the prescribed conditions and h vanishes on the constraint sets (Lagaris et al., 1997). Recent advances make this strategy practical on complex geometries via smooth signed/approximate distance functions, R-functions, and transfinite interpolation, enabling exact Dirichlet enforcement and robust handling of mixed boundary types (Sukumar & Srivastava, 2022; Wang et al., 2023). Relative to penalty-based ("soft") formulations, hard-constraint designs reduce loss-weight tuning, improve optimization conditioning, and often train faster and more stably, while remaining compatible with data terms and inverse problems (Berrone et al., 2023; Barschkis, 2023). Extensions cover Neumann/Robin conditions (via tailored derivative constraints or energetic forms), high-order PDEs (first-order reformulations), and time-dependent problems (time-factorized ansätze), establishing hard-constraint PINNs as a strong default when admissible trial spaces can be constructed (Gladstone et al., 2022; Chen et al., 2025). In contrast to these approaches, B-EPGP exactly satisfies both boundary conditions *and* the PDEs.

Neural operators (Kovachki et al., 2023), such as DeepONets (Lu et al., 2021) and Fourier Neural Operators (FNOs) (Li et al., 2021), represent mappings between function spaces and are trained across families of PDEs. Most neural operators face difficulties enforcing hard boundary constraints, especially in high-dimensional or complex geometries (Göschel et al., 2025; Mousavi et al., 2026). Recent contributions have tackled multi-scale problems (Fang et al., 2023) and high-frequency regimes (Raonic et al., 2024). A growing line of work targets this boundary-condition gap directly: Göschel et al. (2025) generalize local solution structures and introduce an orthogonal-projection approach for strong Neumann/Robin enforcement on piecewise-smooth domains; Mousavi et al. (2026) learn latent pseudo-extensions of boundary data so standard operator backbones respect mixed and variable boundary conditions; and Stuart & Trautner (2026) constrain kernel-integral neural-operator layers to span homogeneous-Dirichlet Laplacian eigenfunctions, proving universal approximation under exact Dirichlet enforcement. Unlike B-EPGP, none of these constructions enforces the governing PDE exactly. Neural operators are connected to GPs (Magnani et al., 2024) via linearization. The approaches of Hennig et al. (2015); Pförtner et al. (2022); Schober et al. (2014); Zhang et al. (2025) improve numerical differential equation solvers by casting them probabilistically.

GP models have been applied to forward and inverse problems involving differential equations. Classical approaches encode *linear* differential constraints directly in covariance functions or through linear operator identities (Macêdo & Castro, 2008; Solin et al., 2018; Särkkä, 2011), including latent-force and state-space formulations (Alvarez et al., 2009; Hartikainen & Sarkka, 2012). Within

this line, EPGP constructs priors from an explicit parametrization of the global solution space of a linear constant-coefficient PDE system (Harkonen et al., 2023), building on earlier parametrization sketches (Jidling et al., 2018) and algorithmic constructions via Gröbner bases (Lange-Hegermann, 2018), with further theoretical characterizations (Henderson et al., 2023a; Sullivan, 2024).

Boundary conditions for GP priors have been incorporated by modifying the prior (e.g., scaling) (Lange-Hegermann, 2021; Lange-Hegermann & Robertz, 2022), and related special cases include BCGP (Dalton et al., 2024b). Most recently, adaptive loss-balancing combined with exact Dirichlet/periodic enforcement (Nguyen et al., 2026) and boundary-condition-guaranteed Kolmogorov–Arnold architectures with radial-basis-function bases (Kim et al., 2025) extend hard-constraint enforcement beyond the MLP trial-solution setting used before. In contrast, B-EPGP enforces homogeneous linear boundary conditions by restricting the Ehrenpreis–Palamodov feature space, yielding priors (and posteriors) that satisfy the PDE and the enforced boundary conditions by construction.

Well-posed boundary value problems where only observations of the source term exist and no observations of the solution have been considered with GPs using spectral expansions of covariance functions (Gulian et al., 2022), including Lie symmetries (Dalton et al., 2024a). EPGP has recently been generalized to inverse problems (Li et al., 2025) and to eigenvalue problems of linear operators via posterior-covariance detection (Bai & Yang, 2026). Beyond constant-coefficient linear settings, GP-based PDE methods often rely on collocation or likelihood constructions. For example, probabilistic numerics and meshless GP solvers propagate discretization uncertainty in PDE-constrained inference (Cockayne et al., 2017), and collocation-based GP frameworks can treat nonlinear PDEs (Chen et al., 2021). Related likelihood- and manifold-based approaches include PIGP and AutoIP (Li et al., 2024; Long et al., 2022). B-EPGP is complementary: it targets exact, analytic enforcement for linear constant-coefficient PDE with linear boundary conditions. Physics-informed state-space GPs such as PHYSS-GP (Hamelijnck et al., 2024) and latent-force or probabilistic-exponential integrator models (Alvarez et al., 2009; Hartikainen & Sarkka, 2012; Bosch et al., 2023) likewise incorporate linear differential operators into GP models, typically via residual terms, state-space constructions, or problem-specific constraints. In contrast to these collocation- or likelihood-based approaches, B-EPGP directly constructs GP priors whose sample paths satisfy both a given constant-coefficient linear PDE system and its boundary conditions exactly, by combining the Ehrenpreis–Palamodov representation of the global solution space with linear boundary operators.

Deterministic PDE solvers (e.g., finite elements and finite differences) are the standard tools for well-posed boundary/initial value problems at prescribed resolution, including finite element methods (Larson & Bengzon, 2013; Logan, 2011; Johnson, 2009; Brenner, 2008; Braess, 2001), finite difference methods (Ervedoza & Zuazua, 2012; Zuazua, 2005; Langtangen & Linge, 2016; 2017; Thomas, 2013), and both (Cohen & Pernet, 2017; Mazumder, 2015; Ames, 2014; Esfandiari, 2017). Our focus is complementary: we construct *probabilistic* models for regression and uncertainty quantification over PDE solution sets from partial observations, where embedding the PDE and boundary constraints into the prior is the key modeling choice. Amo et al. (2026) showed that EPGP surrogates used as deterministic solvers can be several orders of magnitude more accurate than finite elements at equal degrees of freedom.

4 THE B-EPGP METHOD

In both numerical and statistical literature, instances where approximants of the solutions of BVP satisfy the equations exactly are rare. In what follows, we refer to the first equation in BVP as *the PDE* and to the second one as *the BC*. Several works incorporate the PDE as an exact constraint on the prediction (Henderson et al., 2023b; Lange-Hegermann, 2018); of these, Harkonen et al. (2023) introduced a method to generate exact solutions of any PDE, which does not treat boundary conditions. Their main idea is to use the insight of the Ehrenpreis–Palamodov Theorem 2, which generically states that solutions to $A(\partial)u = 0$ in $\mathbb{R}^n$ can be approximated with linear combinations of exponentials

$$u(x) \approx \sum_{j=1}^{M} w_j \exp(x^\top z_j) \quad \text{where } A(z_j) = 0. \tag{EPGP}$$

It is then shown in Harkonen et al. (2023) how to use Gaussian processes to either generate or approximate solutions from data $u(x_i)$, by marginalizing the weights[1] $w_j \in \mathbb{C}$ analytically under their Gaussian prior, and training the frequencies $z_j \in \mathbb{C}^n$. Note that each exponential solves the PDE $A(\partial)\exp(x^\top z) = 0$ if and only if $A(z) = 0$. We will carry out explicit calculations when presenting our examples in Sections 4.2, 5, 6, A. Thus understanding the *characteristic variety* $V = \{z \in \mathbb{C}^n \colon A(z) = 0\}$ is crucial for the implementation. Tools of commutative and computer algebra were used (Cid-Ruiz et al., 2021; Cid-Ruiz & Sturmfels, 2023; Chen & Cid-Ruiz, 2022), particularly the `solvepde` command in Macaulay2 (Chen et al., 2023).

We propose the following strategy: Try to find a linear combination EPGP which satisfies the boundary condition $B(\partial)u = 0$ on $\mathbb{R}^{n-1} \times \{0\}$. Writing $x = (x', x_n)$, we obtain at $x = (x', 0)$

$$\sum_{j=1}^{M} B(z_j) w_j \exp(x'^\top z'_j) = 0.$$

By linear independence of distinct exponentials, we have that all z_js with the same $z'_j = z' \in \mathbb{C}^{n-1}$ must satisfy the restriction

$$\sum_{j\colon z'_j = z'} B(z_j) w_j = 0.$$

This leads us to define the *boundary characteristic variety*

$$S_{z'} = \{(z', z_n) \in \mathbb{C}^n \colon A(z', z_n) = 0\} \quad \text{for } z' \in \mathbb{C}^{n-1},$$

which is generically a finite set. We get a set of functions which satisfy both PDE and BC in BVP:

$$b(x; z') = \sum_{z \in S_{z'}} w_z \exp(x^\top z) \quad \text{for } w_z \text{ such that } \sum_{z \in S_{z'}} B(z) w_z = 0, \tag{B-EP}$$

which are determined up to non-zero multiples. Each **B-EPGP basis** function $b(\,\cdot\,; z')$ solves both equations in BVP exactly. In all our examples we will show that all solutions u of BVP can be approximated with realizations of Gaussian Processes

$$u(x) \approx \sum_{j=1}^{M} w_j b(x; z_j) \quad \text{with } b \text{ defined by B-EP} \tag{B-EPGP}$$

and weights w_j that are independent Gaussian random variables. The posterior distribution gives hard-constrained estimators and its hyperparameters can be estimated by minimizing (twice) the negative log-likelihood up to constants

$$L(Z, \sigma^2, \sigma_0^2) = \tfrac{1}{\sigma_0^2}\big(\|Y\|^2 - Y^\top \Phi G^{-1} \Phi^\top Y\big) + (N - M)\log\sigma_0^2 + \log\det\Sigma + \log\det G,$$

for weights $w_j \sim \mathcal{N}(0, \sigma_j^2/M)$, $\Sigma = \operatorname{diag}(\sigma_j^2)_{j=1}^M$, data $y_h = u(x_h) + \varepsilon_h$ with noise $\varepsilon_h \sim \mathcal{N}(0, \sigma_0^2)$, $h = 1, \ldots, N$, $Y = (y_h)_h$, design matrix $\Phi = (b(x_h; z_j))_{h,j} \in \mathbb{R}^{N \times M}$, and $G = M\sigma_0^2\Sigma^{-1} + \Phi^\top\Phi$. The details are provided in Appendix C.3.

This B-EPGP basis is explicitly computable from the algebraic relation in B-EP and is computed analytically one time. In particular, computation of the B-EPGP basis does not contribute towards training time; see Appendix C.2 for the algebraic details and cost.

This idea is not present in either the algebraic work (Cid-Ruiz et al., 2021; Cid-Ruiz & Sturmfels, 2023; Chen & Cid-Ruiz, 2022; Chen et al., 2023) or in the EPGP paper (Harkonen et al., 2023).

4.1 Examples and theoretical result

Our strategy is computable for any pair of operators A, B in BVP. We present empirical results demonstrating that our method can reconstruct solutions for elliptic, parabolic, and hyperbolic equations in at least three variables, while considering both Dirichlet and Neumann boundary conditions. It is rare that a single method can indiscriminately deal with PDEs of each of these types.

[1] Almost all our examples below concern real-valued functions and operators with real coefficients, so our parameters z_j are actually real in practice and we use real Gaussian Processes in our algorithms. However, the entire presentation of the method uses complex numbers for simplicity of notation; see also Appendix A.3.1.

The PDEs we consider are standard examples that each present typical properties in their respective class: (i) Laplace $A(\partial) = \partial_1^2+\partial_2^2+\ldots+\partial_n^2$, which is elliptic; (ii) heat $A(\partial) = \partial_1-\partial_2^2-\partial_3^2-\ldots-\partial_n^2$, which is parabolic; (iii) wave $A(\partial) = \partial_1^2 - \partial_2^2 - \partial_3^2 - \ldots - \partial_n^2$ which is hyperbolic. The boundary conditions we consider are Dirichlet $B(\partial)u = u$ and Neumann $B(\partial)u = \partial_n u$ on the hyperplane $\mathbb{R}^{n-1} \times \{0\} = \{x \in \mathbb{R}^n \colon x_n = 0\}$.

We moreover provide a theoretical justification showing that realizations of B-EPGP are dense in the set of all solutions of BVP. Our unified statement covers all examples enumerated above:

Theorem 1 (Universal Approximation for B-EPGP)**.** *Suppose that $A(\partial) = A_0(\partial') + \partial_n^2$ and that $B(\partial)u = u$ or $B(\partial)u = \partial_n u$ in BVP. Then any solution $u \in C^\infty(\mathbb{R}^{n-1} \times [0,\infty))$ of BVP can be approximated in the Fréchet topology by functions of the form B-EPGP.*

Approximation in the Fréchet topology, defined in Appendix C.1, is very strong, for instance stronger than uniform or L^2-approximation. We prove Theorem 1 in Appendix C.4. We expect the result of Theorem 1 to hold for any BVP.

4.2 Diverse examples: heat/wave/Laplace PDE with Dirichlet/Neumann BC

We demonstrate the calculation of the B-EPGP basis for the simple example of equation BVP:

$$\begin{cases} \partial_1^2 u - \partial_2^2 u = 0 & \text{in } \mathbb{R} \times (0,\infty) \\ u(x_1, 0) = 0 & \text{for } x_1 \in \mathbb{R}. \end{cases} \tag{Wave 1D}$$

In this case the characteristic variety is $V = \{z \in \mathbb{C}^2 \colon z_1^2 - z_2^2 = 0\} = \{(z', \pm z') \colon z' \in \mathbb{C}\}$. We apply the formula equation B-EPGP: For $z' \in \mathbb{C}^{n-1} = \mathbb{C}$, we have $S_{z'} = \{(z', z'), (z', -z')\}$ and the restriction in B-EP gives

$$b(x_1, x_2; z') = \exp(z'(x_1 - x_2)) - \exp(z'(x_1 + x_2)) \quad \text{for } z' \in \mathbb{C}.$$

In our implementation we make the ad hoc restriction that $z' = \sqrt{-1}\omega$ is purely imaginary, leading after taking real parts to the basis

$$b_1(x_1, x_2; \omega) = \sin(\omega x_1)\sin(\omega x_2), \quad b_2(x_1, x_2; \omega) = \cos(\omega x_1)\sin(\omega x_2), \quad \text{for } \omega \in \mathbb{R}.$$

We use this to construct a design matrix which enables us to perform Gaussian process regression on the following data: Take a true solution of system Wave 1D; we choose $u(x_1, x_2) = 3x_1^2 x_2 + x_2^3$, as it is completely unlike the basis we work with. This demonstrates the generalizability capabilities of the B-EPGP predictions. We sample 1000 data points $(x_{i1}, x_{i2}, u(x_i))$ with $x_i \in [0,1]^2$ on which we train predictions B-EPGP with 500 terms. High accuracy is achieved after short training. The results are presented in Figure 2 and calculation of the basis is presented in Appendix B.1.

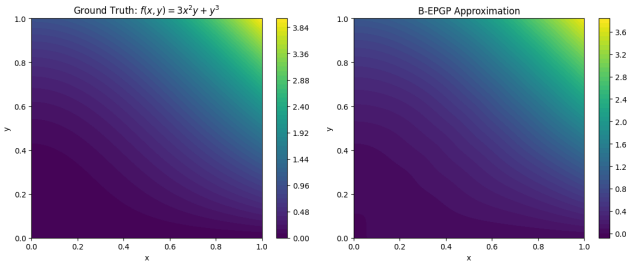


Figure 2: B-EPGP method to approximate the function $u(x, y) = 3x^2y + y^3$ with exact solutions of system Wave 1D. We sampled 1000 data points from $[0,1]^2$ and 500 features in B-EPGP, reaching under 1% validation relative RMSE in under 1 minute of training on a single T4 GPU.

The construction showcased for equation Wave 1D generalizes to other typical—analytically very different—PDEs in higher dimensions and to various boundary conditions. We turn them into a precursory experiment with result presented in Table 1. Details are provided in Appendix B.2. To demonstrate the range of our method, in Appendix A.3 we include examples of reconstructions for systems of PDEs and BCs (multiple equations, multiple conditions, vector-valued unknowns), as well as examples of PDEs with complex coefficients and complex-valued solutions.

Table 1: Performance of B-EPGP for heat, wave, and Laplace equations with Dirichlet or Neumann boundary conditions at $x_3 = 0$. Here we report validation relative RMSE after training on 2,000 data points with 500 features on a single T4 GPU. Training times are all under 1 minute.

PDE		**Dirichlet BC**	**Neumann BC**
$\partial_1 - \partial_2^2 - \partial_3^2$	Ground Truth	$2x_1x_3 + x_2^2x_3$	$2x_1x_2 + x_2x_3^2$
	Error	0.59%	0.72%
$\partial_1^2 - \partial_2^2 - \partial_3^2$	Ground Truth	$x_1^2x_3 + x_2^2x_3$	$x_1^2x_2 + x_2x_3^2$
	Error	0.38%	0.55%
$\partial_1^2 + \partial_2^2 + \partial_3^2$	Ground Truth	$x_1^2x_3 - x_2^2x_3$	$x_1x_2^2 - x_1x_3^2$
	Error	0.91%	0.63%

Table 2: The mean relative L_1 validation error in $[0, 8] \times [0, 4]$ for the experiment Wave IBVP shows that B-EPGP achieves best accuracy against six state-of-the-art baselines trained on the same 1,200 data points. B-EPGP is the only method which satisfies both PDE and BC analytically.

Algorithm	PDE residual (L_1)	BC residual (L_1)	Relative L_1 Error
CNO	(6.21 ± 0.18)e-4	(3.55 ± 0.12)e-4	$0.79 \pm 0.12\%$
FNO	(4.82 ± 0.74)e-3	(2.75 ± 0.42)e-3	$1.17 \pm 0.18\%$
WIGPR	**0**	(2.34 ± 0.05)e-5	$0.21 \pm 0.03\%$
EPGP	**0**	(3.15 ± 0.06)e-5	$0.14 \pm 0.02\%$
B-EPGP (ours)	**0**	**0**	$\mathbf{0.06 \pm 0.01\%}$
BCGP	(1.45 ± 0.21)e-4	**0**	$0.11 \pm 0.02\%$
HC-PINNs	(5.32 ± 0.88)e-4	**0**	$0.43 \pm 0.08\%$

5 B-EPGP AS A NUMERICAL SOLVER FOR WELL-POSED BVPS

In the deterministic study of science, it is typical to examine idealized problems in which data uniquely determines the solution. For example, if we provide knowledge of u and $\partial_1 u$ at $x_1 = 0$ (initial condition and initial velocity), the solution u exists and is uniquely determined by these conditions. Here we will discuss the case of Neumann boundary conditions $\partial_2 u = 0$ at $x_2 = 0$:

$$\begin{cases} \partial_1^2 u - \partial_2^2 u = 0 & \text{in } (0, \infty)^2 \\ \partial_2 u(x_1, 0) = 0 & \text{for } x_1 > 0 \\ u(0, x_2) = h_1(x_2) \text{ and } \partial_1 u(0, x_2) = h_2(x_2) & \text{for } x_2 > 0. \end{cases} \tag{Wave IBVP}$$

Given $h_1, h_2 \colon \mathbb{R} \to \mathbb{R}$, the solution $u \colon (0, \infty)^2 \to \mathbb{R}$ of the Initial-Boundary Value Problem Wave IBVP is known to be uniquely determined. For our algorithm, this means that we only sample data from a 1D space $\{x_1 = 0\}$ and predict the solution in 2D using the B-EPGP basis.

We compare with CNO (Raonic et al., 2024), FNO (Li et al., 2021), WIGPR (Henderson et al., 2023b), EPGP (Harkonen et al., 2023), BCGP (Dalton et al., 2024b), HC-PINNs (Sukumar & Srivastava, 2022) for accuracy starting with the same data. Of these, the NOs, WIGPR, and EPGP do not satisfy BCs exactly, whereas BCGP and HC-PINNs do not satisfy the PDEs exactly. The exact enforcement enables B-EPGP to attain the best accuracy compared to the ground truth

$$u(x_1, x_2) = f(x_2 + x_1 - 3) + f(x_2 - x_1 + 3) + g(x_2 + x_1 - 1) + g(x_2 - x_1 - 1) \\ + g(x_2 + x_1 + 1) + g(x_2 - x_1 + 1) \text{ where } f(x) = e^{-5x^2},\ g(x) = e^{-10x^2}/2.$$

We use this solution to calculate h_1, h_2, which we use to specify data only at $x_1 = 0$. The errors to ground truth are presented in Table 2. B-EPGP is the only method which achieves null PDE and BC residuals. We present other experimental details in Appendix B.1, where we also present the results achieved by the seven methods with fewer data points (Table 4).

5.1 HIGHER DIMENSIONAL PDES, SEVERAL BOUNDARY CONDITIONS

We also study an Initial-Boundary Value problem for the wave equation in three space dimensions (a 4D PDE). We show that B-EPGP can approximate a uniquely determined solution while enforcing

boundary conditions on three hyperplanes simultaneously:

$$\begin{cases} \partial_1^2 u - \partial_2^2 u - \partial_3^2 u - \partial_4^2 u = 0 & \text{for } x_1, x_3, x_4 > 0,\ x_2 \in \mathbb{R} \\ \partial_1 u = 0,\ \partial_3 u = 0,\ \partial_4 u = 0 & \text{respectively at } x_1 = 0,\ x_3 = 0,\ x_4 = 0. \\ u = e^{-5r_1^2} + e^{-5r_2^2} + e^{-5r_3^2} & \text{at } x_1 = 0, \end{cases} \quad \text{(Wave 3D)}$$

where r_1, r_2, r_3 denote the distances between (x_2, x_3, x_4) and $(1,1,1)$, $(1,-1,1)$, $(1,1,-1)$ respectively. Extending the computations in Section 4.2, we obtain the B-EPGP basis

$$\cos(z_1 x_1)\cos(z_2 x_2)\cos(z_3 x_3)\cos(z_4 x_4),\ \cos(z_1 x_1)\sin(z_2 x_2)\cos(z_3 x_3)\cos(z_4 x_4),$$

where $z_1 = \sqrt{z_2^2 + z_3^2 + z_4^2}$ for $z_2, z_3, z_4 \in \mathbb{R}$. We compare with the true solution in Figure 3.

Space-time formulations of the 3D wave equation lead to four input variables. In this regime, grid-based discretizations and boundary-collocation baselines can become memory-limited due to the polynomial growth of boundary samples and the resulting design matrices. In our setup, EPGP exceeds the memory budget on an Nvidia A100 (80GB), while B-EPGP avoids collocation points at $x_1 = 0$, $x_3 = 0$, $x_4 = 0$, and achieves an absolute L_1 error of 8.8×10^{-4}. The wall-clock time to learn frequencies for B-EPGP in this experiment is 2371 seconds; details are in Appendix B.3.

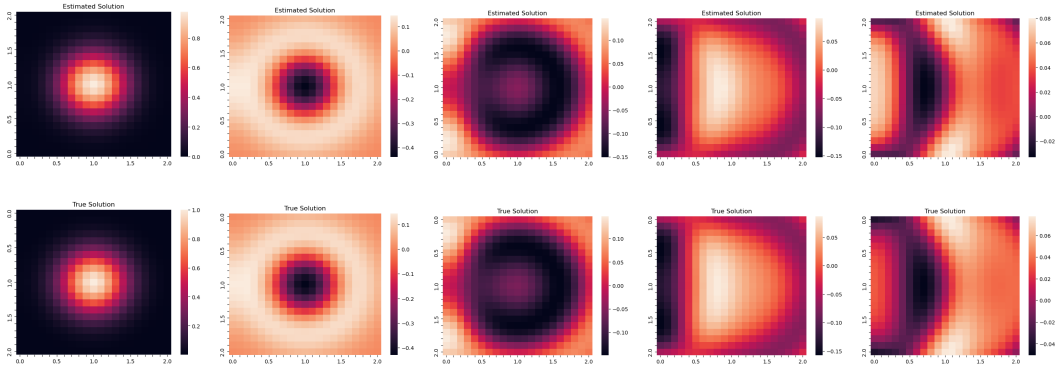


Figure 3: Visual comparison between B-EPGP prediction and true solution for Wave 3D at $x_1 = 0.0, 0.5, 1.0, 1.5, 2.0$, $x_3 = 1.0$, and $x_2, x_4 \in [0, 2]$.

In Appendix A.2, we give an example for the 2D heat equation with BCs on several hyperplanes.

5.2 Examples in polyhedral domains

In Section 5 we covered an unbounded domain bounded by hyperplanes and we compared our method with other probabilistic methods. Here we give an example in which we compare our method with grid-based deterministic numerical solvers. We look at

$$\begin{cases} \partial_1 u - D(\partial_2^2 u + \partial_3^2 u) = 0 & \text{in } [0,1]^3 \\ u = 0 & \text{at } x_2 \text{ or } x_3 \in \{0,1\},\ x_1 \in [0,1] \\ u(0, x_2, x_3) = f(x_2, x_3) & \text{for } x_2, x_3 \in [0,1], \end{cases} \quad \text{(Heat IBVP)}$$

where $D = 0.0125$ is a diffusion coefficient. We consider

$$f(x_2, x_3) = 50/(3\pi)\exp(-200/3((x_2 - 0.5)^2 + (x_3 - 0.5)^2)),$$

a Gaussian pulse which diffuses over time. The B-EPGP basis can be computed explicitly:

$$\sin(j\pi x_2)\sin(k\pi x_3)\exp\left(-x_1\pi^2(j^2 + k^2)/80\right), \quad \text{for } j, k \in \mathbb{N}, \tag{1}$$

which recovers the classical separation of variables method. Since the parameters above are integers, we do not train over them, instead perform least squares optimization to fit parameters w in equation B-EPGP. For the idea behind computing equation 1, see Appendix A.1.1.

B-EPGP took 0.0128s to reach 0.2502% validation RMSE at 20,000 points in $[0,1]^3$ using 144 basis functions 1. We compare to the Forward Time Centered Space (FTCS) finite difference method from

Appendix B.5. FTCS took 0.0323s to reach 0.4522%. Both methods satisfy the BCs exactly, but FTCS has a PDE residual of 4.46e-13, unlike B-EPGP which satisfies the PDE exactly as well.

In Appendix A.1, we construct B-EPGP bases for the 2D wave equation in both rectangular and triangular domains and approximate the unique solution of an IBVP for each.

6 Data-driven predictions from sparse data

The Helmholtz equation with Robin boundary conditions is used to model acoustic impedance (Anderssohn et al., 2006). A practical problem is to reconstruct the acoustic pressure using a few microphones. We idealize this problem using very few data points from a true solution of

$$\begin{cases} \partial_1^2 u + \partial_2^2 u + \partial_3^2 u + u = 0 & \text{in } [0,1]^3 \\ \partial_3 u + u = 0 & \text{on } x_3 = 0. \end{cases} \tag{Helmholtz 3D}$$

and the ground truth $u(x_1, x_2, x_3) = (1 - x_3) J_0(\sqrt{x_1^2 + x_2^2})$, where J_0 is the Bessel function of the first kind of order zero. We take as few as $10, 20, 30, 40, 50$ data points $(x, u(x))$ and obtain relative validation RMSE under 0.3% in seconds, see Table 3 using 600 B-EPGP basis elements, i.e.,

$$\cos(x_1 z_1 + x_2 z_2)(z_3 \cos(x_3 z_3) - \sin(x_3 z_3)),\ \sin(x_1 z_1 + x_2 z_2)(z_3 \cos(x_3 z_3) - \sin(x_3 z_3)),$$
$$\text{where } z_1 = \sin\theta\cos\varphi,\ z_2 = \sin\theta\sin\varphi,\ z_3 = \cos\theta \text{ for } \theta \in [0, \pi],\ \varphi \in [0, 2\pi].$$

None of the other state-of-the-art methods can operate with so little input since they would at least require a large number of collocation points for either the PDE or the BC; details in Appendix B.4.

Table 3: Reconstruction error with extremely sparse data for the Helmholtz 3D problem.

Number of data points	50	40	30	20	10
Validation relative RMSE	0.288%	0.262%	0.297%	0.204%	0.211%

7 Conclusions and limitations

In this paper, we introduce Boundary Ehrenpreis–Palamodov Gaussian Processes (B-EPGPs), a novel probabilistic framework that embeds both linear constant-coefficient PDEs and linear boundary conditions directly into the Gaussian process prior. By analytically restricting the Ehrenpreis–Palamodov representation, our method guarantees that every sample path exactly satisfies both the governing physical equations and boundary constraints by construction.

Empirically, our extensive experiments across elliptic, parabolic, and hyperbolic systems demonstrate that B-EPGP improves upon state-of-the-art predictive accuracy while maintaining strict zero PDE and boundary condition residuals. Compared to existing probabilistic baselines, neural operators, and penalty-based physics-informed neural networks, B-EPGP converges more accurately and completely avoids the optimization trade-offs inherent to soft constraints. Furthermore, our formulation scales successfully to higher-dimensional problems where prior algebraic GP methods exhaust memory limits, and it exhibits exceptional data efficiency, accurately reconstructing complex systems like the 3D Helmholtz equation with as few as ten sparse observations. B-EPGP's versatility covers complex-valued formulations, such as the 1D Schrödinger equation, and coupled systems of PDEs, including the 3D curl and Maxwell's equations, presented in the supplementary material.

Theoretically, we rigorously established the foundational validity of our approach by proving a universal approximation theorem for B-EPGPs. Specifically, we demonstrated that for half-space Dirichlet and Neumann boundary value problems of a large class of PDEs, the realizations of our restricted Gaussian process priors are dense in the Fréchet topology of the exact solution space. This guarantees that any smooth solution to the specified physical system can be approximated to arbitrary precision by our algebraically constructed feature maps, offering a robust mathematical guarantee that complements the method's strong empirical performance.

Limitations. B-EPGP deals with linear PDE constraints and linear boundary conditions with constant coefficients on polyhedral domains. In particular, we do not consider nonlinear equations, material inhomogeneities, or curved boundaries. Currently B-EPGP cannot be used to generate solutions or as a solver of inhomogeneous equations $A(\partial)u = f$ with boundary conditions $B(\partial)u = g$.

AI USE STATEMENT

We used generative AI tools for creating or editing software code, sourcing/searching for information, editing the research paper to improve readability, and identifying relevant literature. We have reviewed all AI-assisted work. LLM-generated code was verified and tested for correctness by all authors. The whole text was proofread by all authors to ensure LLM-generated edits did not introduce artifacts. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI. We have not used generative AI tools to generate synthetic data sets, develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proofs, assist in writing proofs, propose or refine hypotheses, or design or give feedback on the research methodology or experiments. Translation, cleaning or reformatting of datasets, qualitative or thematic data analysis, and interpretation of results with AI assistance do not apply to this work.

REPRODUCIBILITY STATEMENT

To ensure the full reproducibility of our empirical and theoretical results, we have provided comprehensive documentation and resources throughout the manuscript and supplementary materials. The complete source code (including the Python scripts for Gaussian process optimization) is included in the supplementary material, alongside the animations referenced in the text. Detailed experimental setups, including dataset generation, hyperparameter choices (e.g., learning rates, number of epochs, number of basis functions M, and sample sizes N), and hardware specifications (NVIDIA T4 and A100 GPUs) are thoroughly documented in Appendix A and Appendix B.1 through B.5. Furthermore, the algebraic derivation of our method and the exact loss formulation (negative log marginal likelihood) are detailed in Appendices C.2 and C.3. Finally, the rigorous theoretical guarantee of our method, including the complete proof of the Universal Approximation for B-EPGPs (Theorem 1), is provided in Appendix C.4. All data we used is artificially generated and the formulas used to generate it are included in the description of the experiments.

REFERENCES

Shmuel Agmon, Avron Douglis, and Louis Nirenberg. Estimates near the boundary for solutions of elliptic partial differential equations satisfying general boundary conditions. i. *Communications on pure and applied mathematics*, 12(4):623–727, 1959.

Rida Ait El Manssour, Marc Härkönen, and Bernd Sturmfels. Linear PDE with constant coefficients. *Glasgow Mathematical Journal*, 2021.

Mauricio Alvarez, David Luengo, and Neil D. Lawrence. Latent force models. In *AISTATS*, 2009.

William F Ames. *Numerical methods for partial differential equations*. Academic press, 2014.

Obed Amo, Samit Ghosh, Markus Lange-Hegermann, Bogdan Raita, and Michael Pokojovy. Comparing epgp surrogates and finite elements under degree-of-freedom parity. *The International FLAIRS Conference Proceedings*, 39(1), May 2026. doi: 10.32473/flairs.39.1.141836. URL `https://journals.flvc.org/FLAIRS/article/view/141836`.

R Anderssohn, St Marburg, H-J Hardtke, and Chr Grossmann. Reconstruction of sound pressure field by ifem. In *Theoretical And Computational Acoustics 2005: (With CD-ROM)*, pp. 1–4. World Scientific, 2006.

Tianming Bai and Jiannan Yang. Physics-informed Gaussian process regression in solving eigenvalue problem of linear operators, 2026.

Sebastian Barschkis. Exact and soft boundary conditions in physics-informed neural networks for the variable coefficient poisson equation. *arXiv:2310.02548*, 2023.

Thorsten W Becker, Caludio Faccenna, Richard J O'Connell, and Domenico Giardini. The development of slabs in the upper mantle: Insights from numerical and laboratory experiments. *Journal of Geophysical Research: Solid Earth*, 104(B7):15207–15226, 1999.

Sandro Bellini, Maurizio Piacentini, Ciro Cafforio, and F Rocca. Compensation of tissue absorption in emission tomography. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 27(3): 213–218, 1979.

Stefano Berrone, Claudio Canuto, Moreno Pintore, and N. Sukumar. Enforcing dirichlet boundary conditions in physics-informed neural networks and variational physics-informed neural networks. *Heliyon*, 2023.

Andreas Besginow and Markus Lange-Hegermann. Constraining Gaussian Processes to Systems of Linear Ordinary Differential Equations. In *NeurIPS*. 2022.

Andreas Besginow, Markus Lange-Hegermann, and Jörn Tebbe. Linear ordinary differential equations constrained Gaussian processes for solving optimal control problems. *arXiv:2504.12775*, 2025.

Fischer Black and Myron Scholes. The pricing of options and corporate liabilities. *Journal of Political Economy*, 81(3):637–654, 1973.

Vladimir Igorevich Bogachev. *Gaussian measures*. Number 62. American Mathematical Soc., 1998.

Nathanael Bosch, Philipp Hennig, and Filip Tronarp. Probabilistic exponential integrators. *Advances in Neural Information Processing Systems*, 36:40450–40467, 2023.

Dietrich Braess. *Finite elements: Theory, fast solvers, and applications in solid mechanics*. Cambridge University Press, 2001.

Susanne Brenner. *The mathematical theory of finite element methods*. Springer, 2008.

Jialei Chen, Zhehui Chen, Chuck Zhang, and CF Jeff Wu. Apik: Active physics-informed kriging model with partial differential equations. *SIAM/ASA Journal on Uncertainty Quantification*, 2022.

Justin Chen and Yairon Cid-Ruiz. Primary decomposition of modules: a computational differential approach. *Journal of Pure and Applied Algebra*, 2022.

Justin Chen, Yairon Cid-Ruiz, Marc Härkönen, Robert Krone, and Anton Leykin. Noetherian operators in macaulay2. *Journal of Software for Algebra and Geometry*, 12(1):33–41, 2023.

Yifan Chen, Bamdad Hosseini, Houman Owhadi, and Andrew M Stuart. Solving and learning nonlinear PDEs with Gaussian processes. *Journal of Computational Physics*, 2021.

Zhaolin Chen, Siu-Kai Lai, Zhicheng Yang, Yi-Qing Ni, et al. At-pinn-hc: A refined time-sequential method incorporating hard-constraint strategies for predicting structural behavior under dynamic loads. *Computer Methods in Applied Mechanics and Engineering*, 436, 2025.

Philippe G Ciarlet. *The finite element method for elliptic problems*. SIAM, 2002.

Yairon Cid-Ruiz and Bernd Sturmfels. Primary decomposition with differential operators. *International Mathematics Research Notices*, 2023(14):12119–12147, 2023.

Yairon Cid-Ruiz, Roser Homs, and Bernd Sturmfels. Primary ideals and their differential equations. *Foundations of Computational Mathematics*, pp. 1–37, 2021.

Jon Cockayne, Chris J. Oates, T. J. Sullivan, and Mark Girolami. Probabilistic numerical methods for PDE-constrained Bayesian inverse problems. *AIP Conference Proceedings*, 1853(1):060001, 2017. doi: 10.1063/1.4985359.

Gary Cohen and Sébastien Pernet. *Finite element and discontinuous Galerkin methods for transient wave equations*. Springer, 2017.

David A. Cox, John Little, and Donal O'Shea. *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra*. Undergraduate Texts in Mathematics. Springer, Cham, 4th edition, 2015. doi: 10.1007/978-3-319-16721-3.

David Dalton, Dirk Husmeier, and Hao Gao. Physics and lie symmetry informed Gaussian processes. In *ICML*, 2024a.

David Dalton, Alan Lazarus, Hao Gao, and Dirk Husmeier. Boundary constrained Gaussian processes for robust physics-informed machine learning of linear partial differential equations. *Journal of Machine Learning Research*, 25(272):1–61, 2024b.

Sylvain Ervedoza and Enrique Zuazua. The wave equation: Control and numerics. *Control of Partial Differential Equations: Cetraro, Italy 2010, Editors: Piermarco Cannarsa, Jean-Michel Coron*, pp. 245–339, 2012.

Ramin S Esfandiari. *Numerical methods for engineers and scientists using MATLAB®*. Crc Press, 2017.

Shikai Fang, Madison Cooley, Da Long, Shibo Li, Robert Kirby, and Shandian Zhe. Solving high-frequency and multi-scale PDEs with Gaussian processes. *ICLR*, 2023.

Andreas Geist and Sebastian Trimpe. Learning constrained dynamics with Gauss' principle adhering Gaussian processes. In Alexandre M. Bayen, Ali Jadbabaie, George Pappas, Pablo A. Parrilo, Benjamin Recht, Claire Tomlin, and Melanie Zeilinger (eds.), *Proceedings of the 2nd Conference on Learning for Dynamics and Control*, volume 120 of *Proceedings of Machine Learning Research*. PMLR, 2020.

Rini J. Gladstone, Mohammad A. Nabian, N. Sukumar, Ankit Srivastava, and Hadi Meidani. Fo-pinns: A first-order formulation for physics informed neural networks. *arXiv:2210.14320*, 2022.

Niklas Göschel, Sebastian Götschel, and Daniel Ruprecht. Enforcing boundary conditions for physics-informed neural operators, 2025.

Pierre Grisvard. *Elliptic problems in nonsmooth domains*. SIAM, 2011.

Mamikon Gulian, Ari Frankel, and Laura Swiler. Gaussian process regression constrained by boundary value problems. *Computer Methods in Applied Mechanics and Engineering*, 388:114117, 2022.

Oliver Hamelijnck, Arno Solin, and Theodoros Damoulas. Physics-informed variational state-space Gaussian processes. In *Advances in Neural Information Processing Systems*, 2024.

Marc Harkonen, Markus Lange-Hegermann, and Bogdan Raita. Gaussian process priors for systems of linear partial differential equations with constant coefficients. In *ICML*, 2023.

Jouni Hartikainen and Simo Sarkka. Sequential inference for latent force models. *arXiv:1202.3730*, 2012.

Iain Henderson, Pascal Noble, and Olivier Roustant. Characterization of the second order random fields subject to linear distributional PDE constraints. *arXiv:2301.06895*, 2023a.

Iain Henderson, Pascal Noble, and Olivier Roustant. Wave equation-tailored Gaussian process regression with applications to related inverse problems. *hal-03941939*, 2023b.

Philipp Hennig, Michael A Osborne, and Mark Girolami. Probabilistic numerics and uncertainty in computations. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2179):20150142, 2015.

Carl Jidling, Niklas Wahlström, Adrian Wills, and Thomas B Schön. Linearly Constrained Gaussian Processes. In *NeurIPS*, 2017.

Carl Jidling, Johannes Hendriks, Niklas Wahlström, Alexander Gregg, Thomas B Schön, Christopher Wensrich, and Adrian Wills. Probabilistic Modelling and Reconstruction of Strain. *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, 2018.

Claes Johnson. *Numerical solution of partial differential equations by the finite element method*. Courier Corporation, 2009.

Ehsan Kharazmi, Zhiping Zhang, and George Em Karniadakis. hp-VPINNs: Variational physics-informed neural networks with domain decomposition. *Computer Methods in Applied Mechanics and Engineering*, 374, 2021.

Bongseok Kim, Jiahao Zhang, and Guang Lin. BEKAN: Boundary condition-guaranteed evolutionary kolmogorov-arnold networks with radial basis functions for solving PDE problems, 2025.

Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces with applications to PDEs. *Journal of Machine Learning Research*, 24(89):1–97, 2023.

Isaac E. Lagaris, Aristidis Likas, and Dimitrios I. Fotiadis. Artificial neural networks for solving ordinary and partial differential equations. *arXiv preprint*, 1997.

Markus Lange-Hegermann. Algorithmic Linearly Constrained Gaussian Processes. In *NeurIPS*. 2018.

Markus Lange-Hegermann. Linearly constrained Gaussian processes with boundary conditions. In *AISTATS*, 2021.

Markus Lange-Hegermann and Daniel Robertz. On boundary conditions parametrized by analytic functions. In *Computer Algebra in Scientific Computing*, 2022.

Hans Petter Langtangen and Svein Linge. Finite difference methods for wave equations. *Journal of Numerical Methods*, 2016.

Hans Petter Langtangen and Svein Linge. *Finite difference computing with PDEs: a modern software approach*. Springer Nature, 2017.

Mats G Larson and Fredrik Bengzon. *The finite element method: theory, implementation, and applications*, volume 10. Springer Science & Business Media, 2013.

Miguel Lázaro-Gredilla, Joaquin Quiñonero-Candela, Carl Edward Rasmussen, and Aníbal R. Figueiras-Vidal. Sparse spectrum Gaussian process regression. *JMLR*, 2010.

Xin Li, Markus Lange-Hegermann, and Bogdan Raita. Gaussian process regression for inverse problems in linear PDEs. *arXiv:2502.04276*, 2025.

Zhaohui Li, Shihao Yang, and C. F. Jeff Wu. Parameter inference based on Gaussian processes informed by nonlinear partial differential equations. *SIAM/ASA Journal on Uncertainty Quantification*, 12(3):964–1004, 2024. doi: 10.1137/22M1514131.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *ICLR*, 2021.

Daryl L Logan. *A first course in the finite element method*, volume 4. Thomson, 2011.

Da Long, Zheng Wang, Aditi S. Krishnapriyan, Robert M. Kirby, Shandian Zhe, and Michael W. Mahoney. Autoip: A united framework to integrate physics into Gaussian processes. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 14210–14222, 2022.

Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. volume 3, pp. 218–229. Nature Publishing Group UK London, 2021.

Ives Macêdo and Rener Castro. Learning Divergence-free and Curl-free Vector Fields with Matrix-valued Kernels. *Instituto Nacional de Matematica Pura e Aplicada, Brasil, Tech. Rep*, 2008.

Emilia Magnani, Marvin Pförtner, Tobias Weber, and Philipp Hennig. Linearization turns neural operators into function-valued Gaussian processes. *arXiv:2406.05072*, 2024.

Sandip Mazumder. *Numerical methods for partial differential equations: finite difference and finite volume methods*. Academic Press, 2015.

Peter Monk. *Finite element methods for Maxwell's equations*. Oxford University Press, 2003.

Sepehr Mousavi, Siddhartha Mishra, and Laura De Lorenzis. Imposing boundary conditions on neural operators via learned function extensions, 2026.

Duc Tien Nguyen, Trinh Minh Tuan, Nguyen Duc Manh, Vu Linh Nguyen, and Dinh Gia Ninh. Adaptive hard-soft physics-informed neural networks for robust boundary-constrained PDE solving, 2026.

Marvin Pförtner, Ingo Steinwart, Philipp Hennig, and Jonathan Wenger. Physics-informed Gaussian process regression generalizes linear PDE solvers. *arXiv:2212.12474*, 2022.

Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 2019.

Bogdan Raonic, Roberto Molinaro, Tim De Ryck, Tobias Rohner, Francesca Bartolucci, Rima Alaifari, Siddhartha Mishra, and Emmanuel de Bézenac. Convolutional neural operators for robust and accurate learning of PDEs. *NeurIPS*, 2024.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian processes for machine learning*. MIT Press, Cambridge, MA, 2006.

Simo Särkkä. Linear Operators and Stochastic Partial Differential Equations in Gaussian Process Regression. In *ICANN*. Springer, 2011.

Michael Scheuerer and Martin Schlather. Covariance models for divergence-free and curl-free random vector fields. *Stoch. Models*, 2012.

Michael Schober, David K Duvenaud, and Philipp Hennig. Probabilistic ODE solvers with Runge-Kutta means. *NeurIPS*, 2014.

Peter M Shearer. *Introduction to seismology*. Cambridge university press, 2019.

Arno Solin, Manon Kok, Niklas Wahlström, Thomas B Schön, and Simo Särkkä. Modeling and Interpolation of the Ambient Magnetic Field by Gaussian Processes. *IEEE Transactions on Robotics*, 2018.

Andrew M. Stuart and Margaret Trautner. Enforcing Dirichlet boundary conditions in operator learning, 2026.

N. Sukumar and Ankit Srivastava. Exact imposition of boundary conditions with distance functions in physics-informed deep neural networks. *Computer Methods in Applied Mechanics and Engineering*, 389, 2022. doi: 10.1016/j.cma.2021.114333.

TJ Sullivan. Hille's theorem for bochner integrals of functions with values in locally convex spaces. *Real Analysis Exchange*, 2024.

Peter S Swain, Keiran Stevenson, Allen Leary, Luis F Montano-Gutierrez, Ivan BN Clark, Jackie Vogel, and Teuta Pilizota. Inferring time derivatives including cell growth rates using Gaussian processes. *Nature communications*, 2016.

Jörn Tebbe, Andreas Besginow, and Markus Lange-Hegermann. Physics-informed Gaussian processes as linear model predictive controller. *arXiv:2412.04502*, 2024.

James William Thomas. *Numerical partial differential equations: finite difference methods*, volume 22. Springer Science & Business Media, 2013.

Niklas Wahlström, Manon Kok, Thomas B. Schön, and Fredrik Gustafsson. Modeling Magnetic Fields using Gaussian Processes. In *Proceedings of the 38th International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2013.

Jiaji Wang, Y. L. Mo, Bassam Izzuddin, and Chul-Woo Kim. Exact dirichlet boundary physics-informed neural network (epinn) for solid mechanics. *Computer Methods in Applied Mechanics and Engineering*, 2023. doi: 10.1016/j.cma.2023.116184.

Sifan Wang, Hanwen Wang, and Paris Perdikaris. On the eigenvector bias of fourier feature networks: From regression to solving multi-scale PDEs with physics-informed neural networks. *Computer Methods in Applied Mechanics and Engineering*, 384:113938, 2021.

Jeremy Yu, Lu Lu, Xuhui Meng, and George Em Karniadakis. Gradient-enhanced physics-informed neural networks for forward and inverse PDE problems. *Computer Methods in Applied Mechanics and Engineering*, 393:114823, 2022.

Rui Zhang, Qi Meng, Rongchan Zhu, Yue Wang, Wenlei Shi, Shihua Zhang, Zhi-Ming Ma, and Tie-Yan Liu. Monte carlo neural PDE solver for learning PDEs via probabilistic representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Enrique Zuazua. Propagation, observation, and control of waves approximated by finite difference methods. *SIAM review*, 47(2):197–243, 2005.

A Further experiments

A.1 Polyhedral domains

A.1.1 Rectangular domains

We look for the solution of the initial boundary value problem

$$\begin{cases} \partial_1^2 u - \partial_2^2 u - \partial_3^2 u = 0 & \text{for } x_2, x_3 \in (0,4),\ x_1 \in (0,12) \\ u(x_1, x_2, x_3) = 0 & \text{for } x_2 \text{ or } x_3 = 0 \text{ or } 4. \\ \partial_1 u(0, x_2, x_3) = 0 & \text{for } x_2, x_3 \in (0,4) \\ u(0, x_2, x_3) = \exp(-10((x_2-1)^2 + (x_3-1)^2)) & \text{for } x_2, x_3 \in (0,4). \end{cases}$$

We begin with the B-EPGP basis for the 2D Wave equation with a Dirichlet boundary condition at $x_3 = 0$ from equation 4. We observe that the only linear combination of elements of equation 4 which satisfy the Dirichlet boundary condition also at $x_2 = 0$ are spanned by

$$\cos(z_1 x_1)\sin(z_2 x_2)\sin(z_3 x_3), \quad \sin(z_1 x_1)\sin(z_2 x_2)\sin(z_3 x_3), \quad \text{for } z_1 = \sqrt{z_2^2 + z_3^2}.$$

These satisfy the Dirichlet boundary conditions on $x_2 = 4$ or $x_3 = 4$ precisely when $z_2 = j\pi/4$ and $z_3 = k\pi/4$ for integers j, k. We can moreover incorporate the Neumann boundary condition at $x_1 = 0$ with the B-EPGP basis

$$\cos\left(\frac{\pi}{4}\sqrt{j^2+k^2}\, x_1\right)\sin\left(\frac{\pi}{4} j x_2\right)\sin\left(\frac{\pi}{4} k x_3\right), \quad \text{for } j, k \in \mathbb{N}. \tag{2}$$

That this set of functions has linear span which is dense in the set of all solutions to the three homogeneous equations above follows from the classic separation of variables method.

Snapshots of our reconstruction are presented in Figure 4.

A.1.2 Triangular domains

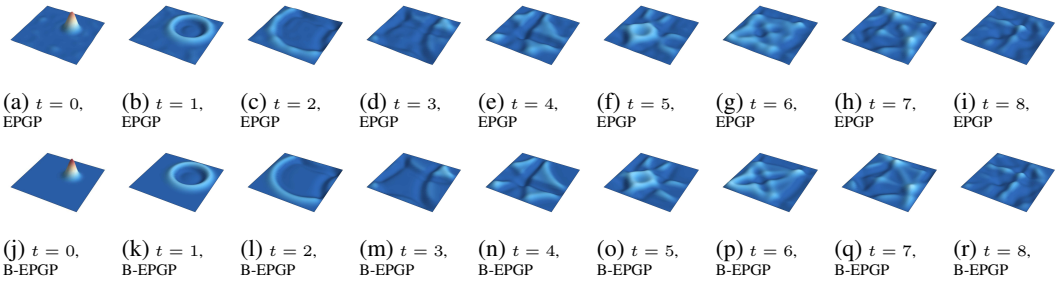
We next consider the triangle $\Omega = \{(x_2, x_3) \colon 0 < x_3 < x_2 < 4\}$ and $x_1 \in (0,4)$ and look for the solution of the initial boundary value problem

$$\begin{cases} \partial_1^2 u - \partial_2^2 u - \partial_3^2 u = 0 & \text{in } \Omega \times (0,4) \\ u = 0 & \text{for } (x_2, x_3) \in \partial\Omega \\ \partial_1 u(0, x_2, x_3) = 0 & \text{in } \Omega \\ u(0, x_2, x_3) = \exp(-10((x_2-1)^2 + (x_3-1)^2)) & \text{in } \Omega \end{cases}$$

Using the B-EPGP basis computed in Section A.1.1 and an odd reflection in the diagonal line $x_2 = x_3$ gives

$$\cos\left(\frac{\pi}{4}\sqrt{j^2+k^2}\, x_1\right)\left[\sin\left(\frac{\pi}{4} j x_2\right)\sin\left(\frac{\pi}{4} k x_3\right) - \sin\left(\frac{\pi}{4} j x_3\right)\sin\left(\frac{\pi}{4} k x_2\right)\right], \quad \text{for } j, k \in \mathbb{N}.$$

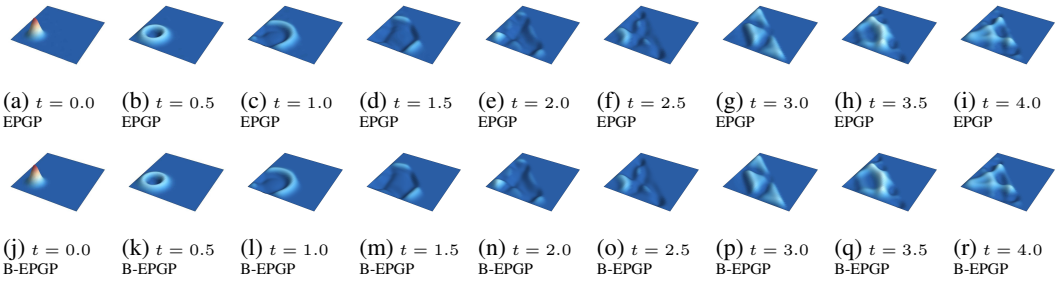
so that this new B-EPGP basis satisfies the Dirichlet boundary condition on all three boundary hyperplanes $\{x_2 = 4\}, \{x_3 = 0\}, \{x_2 = x_3\}$ as well as the Neumann condition on $\{x_1 = 0\}$. Snapshots of our reconstruction are presented in Figure 5.



(a) $t = 0$, EPGP (b) $t = 1$, EPGP (c) $t = 2$, EPGP (d) $t = 3$, EPGP (e) $t = 4$, EPGP (f) $t = 5$, EPGP (g) $t = 6$, EPGP (h) $t = 7$, EPGP (i) $t = 8$, EPGP

(j) $t = 0$, B-EPGP (k) $t = 1$, B-EPGP (l) $t = 2$, B-EPGP (m) $t = 3$, B-EPGP (n) $t = 4$, B-EPGP (o) $t = 5$, B-EPGP (p) $t = 6$, B-EPGP (q) $t = 7$, B-EPGP (r) $t = 8$, B-EPGP

Figure 4: Solution of 2D wave equation in a rectangular domain calculated using the EPGP and B-EPGP methods. We use a Dirichlet boundary condition which is visible above from the fact that the edges of our plots have the same color in all snapshots. The results of B-EPGP seem more reasonable, as they are more stable and lack corrupting high frequencies. Animations can be found in the supplementary material as `rectangle_EPGP.mp4` and `rectangle_B-EPGP.mp4`.



(a) $t = 0.0$ EPGP (b) $t = 0.5$ EPGP (c) $t = 1.0$ EPGP (d) $t = 1.5$ EPGP (e) $t = 2.0$ EPGP (f) $t = 2.5$ EPGP (g) $t = 3.0$ EPGP (h) $t = 3.5$ EPGP (i) $t = 4.0$ EPGP

(j) $t = 0.0$ B-EPGP (k) $t = 0.5$ B-EPGP (l) $t = 1.0$ B-EPGP (m) $t = 1.5$ B-EPGP (n) $t = 2.0$ B-EPGP (o) $t = 2.5$ B-EPGP (p) $t = 3.0$ B-EPGP (q) $t = 3.5$ B-EPGP (r) $t = 4.0$ B-EPGP

Figure 5: Solution of 2D wave equation in a triangular domain with Dirichlet boundary conditions calculated using the EPGP and B-EPGP methods. Animations can be found in the supplementary material as `triangle_EPGP.mp4` and `triangle_B-EPGP.mp4`.

A.2 Reconstructing the physical effects of boundary conditions

We also study the effects of the Dirichlet or Neumann boundary conditions on heat diffusion. We impose these conditions on boundary hyperplanes, $\{x_2 = 0\}$ and $\{x_3 = 0\}$. We begin with the case of Neumann conditions:

$$\begin{cases} \partial_1 u - \partial_2^2 u - \partial_3^2 u = 0 & \text{in } (0,4) \times (0,\infty)^2 \\ \partial_2 u(x_1, 0, x_3) = 0 & \text{for } x_1 \in (0,4),\ x_3 \in (0,\infty) \\ \partial_3 u(x_1, x_2, 0) = 0 & \text{for } x_1 \in (0,4),\ x_2 \in (0,\infty) \\ u(0, x_2, x_3) = 5\exp(-10((x_2-1)^2 + (x_3-1)^2)) & \text{in } (0,\infty)^2. \end{cases}$$

Using the same idea that leads to equation 2, we obtain the B-EPGP basis

$$\exp(-(z_2^2 + z_3^2)t)\cos(z_2 x_2)\cos(z_3 x_3), \quad \text{for } z_2, z_3 \in \mathbb{R}.$$

We report the results on Row 3 of Figure 6, which clearly shows that the initial heat is insulated inside the wedge $\{x_2, x_3 > 0\}$. This is the physical effect produced by the Neumann boundary condition.

For comparison, we will also consider the case of Dirichlet boundary conditions, which which fix the temperature on the boundary of the wedge at $u = 0$. We see in Row 2 of Figure 6 that heat is lost through the boundary, visibly faster than in the case of free diffusion (no boundary condition imposed). The system using the Dirichlet condition is

$$\begin{cases} \partial_1 u - \partial_2^2 u - \partial_3^2 u = 0 & \text{in } (0,4) \times (0,\infty)^2 \\ u(x_1, 0, x_3) = 0 & \text{for } x_1 \in (0,4),\ x_3 \in (0,\infty) \\ u(x_1, x_2, 0) = 0 & \text{for } x_1 \in (0,4),\ x_2 \in (0,\infty) \\ u(0, x_2, x_3) = 5\exp(-10((x_2-1)^2 + (x_3-1)^2)) & \text{in } (0,\infty)^2, \end{cases}$$

for which we obtain the B-EPGP basis

$$\exp(-(z_2^2+z_3^2)t)\sin(z_2x_2)\sin(z_3x_3), \quad \text{for } z_2, z_3 \in \mathbb{R}.$$

In Row 1 of Figure 6 we will plot our approximations of solutions to the heat equation in full space

$$\begin{cases} \partial_1 u - \partial_2^2 u - \partial_3^2 u = 0 & \text{in } (0,4)\times\mathbb{R}^2 \\ u(0,x_2,x_3) = 5\exp(-10((x_2-1)^2+(x_3-1)^2)) & \text{in } \mathbb{R}^2. \end{cases}$$

In this case we use the EPGP basis (Harkonen et al., 2023)

$$\exp(-(z_2^2+z_3^2)t)\cos(z_2x_2)\cos(z_3x_3),\ \exp(-(z_2^2+z_3^2)t)\cos(z_2x_2)\sin(z_3x_3), \quad \text{for } z_2, z_3 \in \mathbb{R}.$$

A.3 Complex coefficients, systems

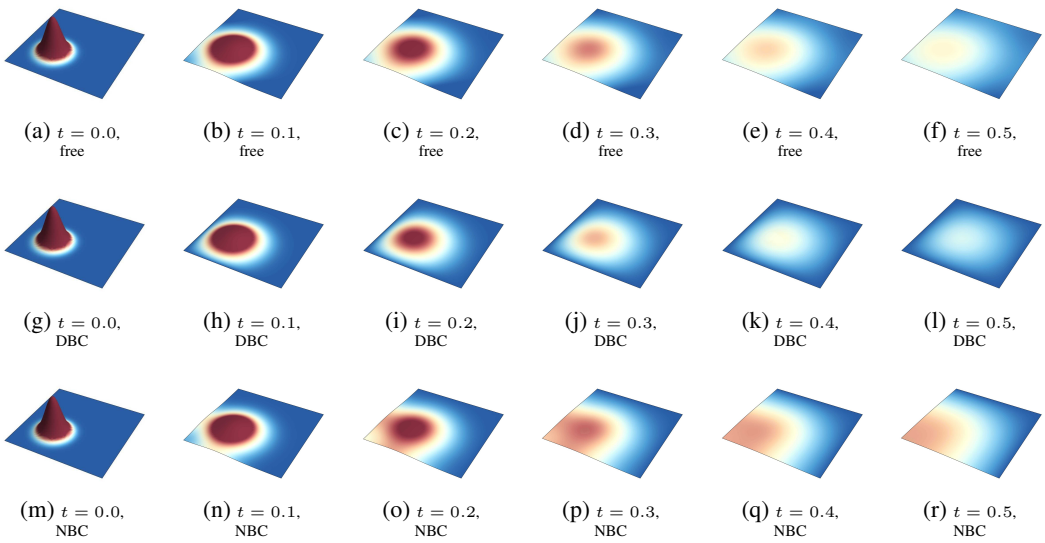
The B-EPGP method extends to systems, meaning considering several PDEs in $\mathbb{R}^{n-1}\times[0,\infty)$ and several BCs on $\mathbb{R}^{n-1}$ for unknowns $u\colon \mathbb{R}^n \to \mathbb{R}^m$. We can moreover consider complex coefficients and complex-valued functions $u\colon \mathbb{R}^n \to \mathbb{C}^m$. Restrictions similar to equation B-EP remain true, with their computation becoming more complicated. Nevertheless, the B-EPGP bases can be computed by more advanced tools of commutative algebra. The details are provided in Appendix C.2.

Here we cover examples of (i) Schrödinger's equation with Dirichlet boundary condition (complex coefficients and complex-valued solutions, Appendix A.3.1); (ii) the curl equation with tangential boundary conditions (3 coupled equations with 2 boundary conditions, Appendix A.3.2); (iii) a reduced Maxwell system with perfect electric conductor boundary conditions (models dynamics of electromagnetic fields, , Appendix A.3.3).

A.3.1 Schrödinger's equation

We consider Schrödinger's equation in one spatial dimension with a Dirichlet boundary condition

$$\begin{cases} \sqrt{-1}\partial_1 u + \partial_2^2 u = 0 & \text{in } \mathbb{R}\times(0,\infty) \\ u(x_1,0) = 0 & \text{for } x_1 \in \mathbb{R}, \end{cases} \qquad \text{(Schrödinger 1D)}$$



(a) $t = 0.0$, free (b) $t = 0.1$, free (c) $t = 0.2$, free (d) $t = 0.3$, free (e) $t = 0.4$, free (f) $t = 0.5$, free

(g) $t = 0.0$, DBC (h) $t = 0.1$, DBC (i) $t = 0.2$, DBC (j) $t = 0.3$, DBC (k) $t = 0.4$, DBC (l) $t = 0.5$, DBC

(m) $t = 0.0$, NBC (n) $t = 0.1$, NBC (o) $t = 0.2$, NBC (p) $t = 0.3$, NBC (q) $t = 0.4$, NBC (r) $t = 0.5$, NBC

Figure 6: Snapshots of our solution for 2D heat equation with different boundary condition at 6 timepoints. The effects of the boundary conditions are very clearly visible. Row 1, free heat equation (no BC): heat diffuses exponentially in time, with radial symmetry in space. Row 2, Dirichlet BC: heat is maintained at 0 on the edges $x = 0$ and $y = 0$ so heat is diffused faster. Row 3, Neumann BC: the edges $x = 0$ and $y = 0$ are thermally insulated, so no heat is diffused there; hence heat is diffused slower and only in directions of increasing x, $y > 0$. Animations can be found in the supplementary material as `heat_free.mp4`, `heat_DBC.mp4`, and `heat_NBC.mp4`.

for functions $u\colon \mathbb{R}^2 \to \mathbb{C}$. In this case we use complex parameters in our parametrization. The characteristic variety of the PDE is given by $\sqrt{-1}z_1 + z_2^2 = 0$ for $z_i \in \mathbb{C}$, leading to the EPGP basis

$$\exp(\sqrt{-1}z^2 x_1 + z x_2) \quad \text{for } z \in \mathbb{C}.$$

This is modified using equation B-EP to obtain the B-EPGP basis

$$b(x_1, x_2; z) = \exp(\sqrt{-1}z^2 x_1 + z x_2) - \exp(\sqrt{-1}z^2 x_1 - z x_2) \quad \text{for } z \in \mathbb{C}.$$

We trained our algorithm on data $(x_1, x_2, x_2^3 + \sqrt{-1}6x_2x_1)$ for 1,000 pairs of points $x_1, x_2 \in [0,1]^2$. In this case we optimize over complex parameters directly and use a CPU to perform the training. We obtained fast convergence to ground truth $x_2^3 + \sqrt{-1}6x_2x_1$ with relative L2 RMSE of 0.01%, see Figure 7 for visuals.

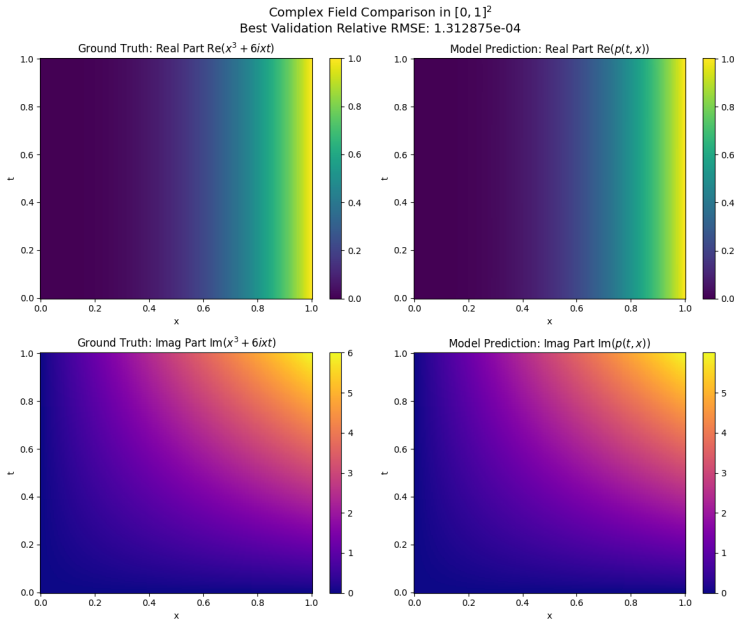


Figure 7: B-EPGP method to approximate the function $u(t,x) = x^3 + 6\sqrt{-1}xt$ with exact solutions of system Schrödinger 1D. Heat maps for the real and imaginary parts of both ground truth and our approximation are plotted.

A.3.2 Curl system, tangential boundary conditions

We consider the curl system of PDEs for vector fields $u\colon \mathbb{R}^3 \to \mathbb{R}^3$ with a tangential boundary condition

$$\begin{cases} \nabla \times u = 0 & \text{in } \{x_3 > 0\} \\ u \times \boldsymbol{n} = 0 & \text{for } x_3 = 0, \end{cases} \tag{Curl 3D}$$

where $\boldsymbol{n}$ denotes the *normal* to a surface in 3D,

$$u \times \boldsymbol{n} = \begin{pmatrix} u_2 n_3 - u_3 n_2 \\ u_3 n_1 - u_1 n_3 \\ u_1 n_2 - u_2 n_1 \end{pmatrix} \quad \text{and} \quad \nabla \times u = \begin{pmatrix} \partial_2 u_3 - \partial_3 u_2 \\ \partial_3 u_1 - \partial_1 u_3 \\ \partial_1 u_2 - \partial_2 u_1 \end{pmatrix}$$

denote the *cross product* and the 3D *curl operator* respectively. To obtain the EPGP basis for the curl system we employ the Macaulay2 command `solvepde`, see Appendices C.1 and C.2 for details and implementation. We obtain the characteristic variety $V = \mathbb{C}^3$ and the Noetherian multiplier $(z_1, z_2, z_3)^\top$ which leads to the EPGP basis

$$(z_1, z_2, z_3)^\top \exp(x^\top z) \quad \text{for } z \in \mathbb{R}^3.$$

In theory we should take $z \in \mathbb{C}^3$, but we observe empirically that real parameters suffice here.

We next break down the boundary condition. The normal to the hyperplane $\{x \in \mathbb{R}^3 \colon x_3 = 0\}$ is clearly $\boldsymbol{n} = (0,0,1)^\top$, which leads to $u \times \boldsymbol{n} = (u_2, -u_1, 0)$. Thus we have two boundary conditions

$$u_1(x_1, x_2, 0) = 0, \quad u_2(x_1, x_2, 0) = 0 \quad \text{for } x_1, x_2 \in \mathbb{R}.$$

Performing the calculations in equation B-EP leads to the B-EPGP basis

$$b(x; z) = \begin{pmatrix} z_1 \\ z_2 \\ z_3 \end{pmatrix} \exp(x_1 z_1 + x_2 z_2 + x_3 z_3) - \begin{pmatrix} z_1 \\ z_2 \\ -z_3 \end{pmatrix} \exp(x_1 z_1 + x_2 z_2 - x_3 z_3) \quad \text{for } z \in \mathbb{R}^3.$$

We tested our algorithm against the ground truth $u(x) = (2x_1x_2^2x_3,\ 2x_1^2x_2x_3,\ x_1^2x_2^2)^\top$ for $x \in \mathbb{R}^3$. We sampled $N = 2{,}000$ points $(x, u(x))$ with $x \in [0,1]^3$, used $M = 500$ basis functions, trained for 72,500 epochs to obtain a relative L_2 error of 5% to the ground truth, see also Figure 8.

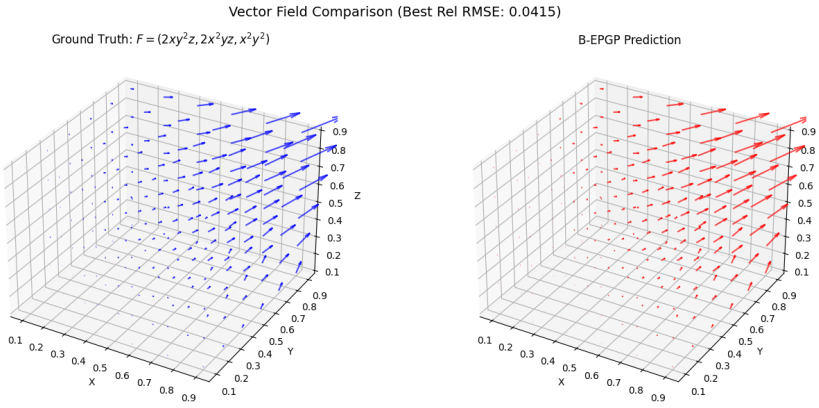


Figure 8: Quiver plots for the B-EPGP reconstruction of a solution of Curl 3D.

A.3.3 Electromagnetic fields under perfect electric conductor conditions

We also consider the classic Maxwell's system with Perfect Electric Conductor boundary conditions:

$$\begin{cases} \partial_0 E - \nabla \times B = 0 & \text{in } x_2 > 0 \\ \partial_0 B + \nabla \times E = 0 & \text{in } x_2 > 0 \\ \nabla \cdot E = 0 & \text{in } x_2 > 0 \\ \nabla \cdot B = 0 & \text{in } x_2 > 0 \\ E \times \boldsymbol{n} = 0 & \text{on } x_2 = 0, \end{cases} \tag{Maxwell 3D}$$

where $E, B \colon \mathbb{R}^4 \to \mathbb{R}^3$, $x = (x_0, x_1, x_2, x_3)$, so that $\nabla \times E$ and $E \times \boldsymbol{n}$ are defined as in Section A.3.2 and moreover $\nabla \cdot E = \partial_1 E_1 + \partial_2 E_2 + \partial_3 E_3$ is the *divergence* operator. We will consider a simplified version of the system, in which the unknown fields are independent of x_3 and

$$B(x) = (B_1(x_0, x_1, x_2), B_2(x_0, x_1, x_2), 0), \quad E(x) = (0, 0, E_3(x_0, x_1, x_2)).$$

We obtain the reduced system for unknown $(B_1, B_2, E_3) \colon \mathbb{R}^3 \to \mathbb{R}^3$

$$\begin{cases} \partial_0 E_3 + \partial_2 B_1 - \partial_1 B_2 = 0 & \text{in } x_2 > 0 \\ \partial_0 B_1 + \partial_2 E_3 = 0 & \text{in } x_2 > 0 \\ \partial_0 B_2 - \partial_1 E_3 = 0 & \text{in } x_2 > 0 \\ \partial_1 B_1 + \partial_2 B_2 = 0 & \text{in } x_2 > 0 \\ E_3 = 0 & \text{on } x_2 = 0. \end{cases} \tag{Maxwell 2D}$$

The `solvepde` command in Macaulay2 yields the characteristic variety $V = \{(z_0, z_1, z_2) \in \mathbb{C}^3 \colon z_0^2 = z_1^2 + z_2^2\}$ and the Noetherian multiplier $(-z_2, z_1, z_0)^\top$, leading to the EPGP basis

$$\begin{pmatrix} -z_2 \\ z_1 \\ z_0 \end{pmatrix} \cos(z^\top x), \ \begin{pmatrix} -z_2 \\ z_1 \\ z_0 \end{pmatrix} \sin(z^\top x) \quad \text{for } z_1, z_2 \in \mathbb{R} \text{ and } z_0 = \pm\sqrt{z_1^2 + z_2^2}.$$

Incorporating the boundary condition gives the B-EPGP basis

$$\begin{pmatrix} -z_2 \\ z_1 \\ z_0 \end{pmatrix} \cos(z_0x_0 + z_1x_1 + z_2x_2) - \begin{pmatrix} z_2 \\ z_1 \\ z_0 \end{pmatrix} \cos(z_0x_0 + z_1x_1 - z_2x_2),$$
$$\begin{pmatrix} -z_2 \\ z_1 \\ z_0 \end{pmatrix} \sin(z_0x_0 + z_1x_1 + z_2x_2) - \begin{pmatrix} z_2 \\ z_1 \\ z_0 \end{pmatrix} \sin(z_0x_0 + z_1x_1 - z_2x_2)$$

for $z_0 = \pm\sqrt{z_1^2 + z_2^2}$ and $z_1, z_2 \in \mathbb{R}$.

We tested our algorithm against the ground truth

$$B_1(x_0, x_1, x_2) = -5x_1x_0^2 - 5/2x_1x_2^2 - 5/6x_1^3$$
$$B_2(x_0, x_1, x_2) = 5x_2x_0^2 + 5/2x_1^2x_2 + 5/6x_2^3$$
$$E_3(x_0, x_1, x_2) = 10x_0x_1x_2.$$

We sampled $N = 5{,}000$ points $(x, B_1(x), B_2(x), E_3(x))$ with $x \in [0, 1]^3$, used $M = 1{,}000$ basis functions, trained for $15{,}000$ epochs to obtain 5% relative L_2 error to the ground truth, see also Figure 9. For better reproducibility, we incorporate the Macaulay2 code that we used to obtain the

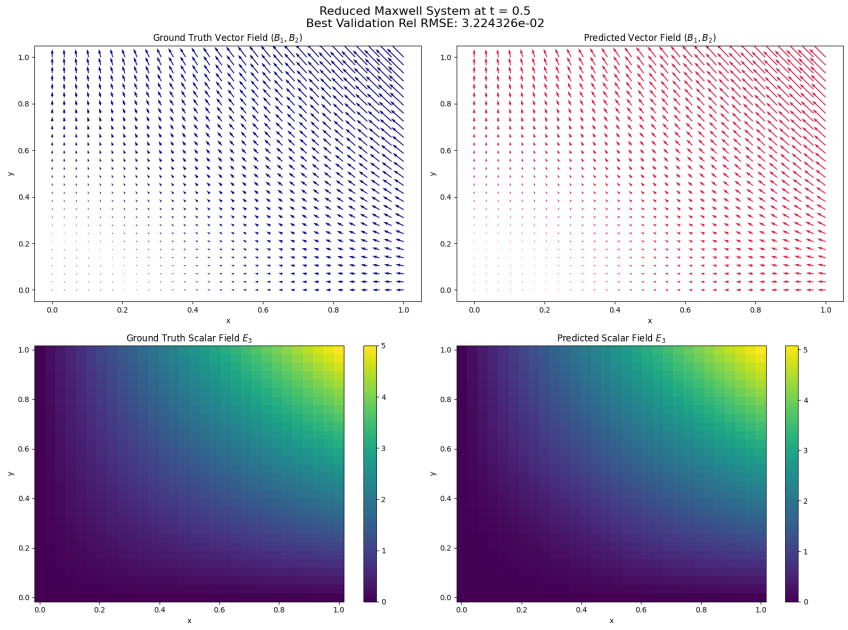


Figure 9: B-EPGP reconstruction of a solution of Maxwell 2D at $t = 0.5$: 2D quiver plot for the magnetic field (B_1, B_2) and heat map for the electric field E_3.

characteristic variety and Noetherian multiplier above:

```
i1 : needsPackage "NoetherianOperators";
i2 : R = QQ[x,y,t];
i3 : M = matrix {
       {t,0,y},
       {0,t,-x},
       {y,-x,t},
       {x,y,0}
     };
i4 : solvePDE transpose M
```

Table 4: The experiment for Wave IBVP was performed with 1,200 data points, see Table 2. Here we report results obtained using 120 data points. B-EPGP is the only method which satisfies both PDE and BC analytically, constructing predictions which satisfy physics constraints truly. This enables our method to converge most accurately to a ground truth.

Algorithm	PDE residual (L_1)	BC residual (L_1)	Relative L_1 Error
CNO	(4.12 ± 0.85)e-3	(2.35 ± 0.45)e-3	$1.31 \pm 0.25\%$
FNO	(1.54 ± 0.32)e-2	(8.91 ± 0.95)e-3	$1.95 \pm 0.35\%$
WIGPR	**0**	(1.42 ± 0.25)e-4	$0.86 \pm 0.09\%$
EPGP	**0**	(1.15 ± 0.18)e-4	$0.52 \pm 0.05\%$
B-EPGP (ours)	**0**	**0**	$\mathbf{0.37 \pm 0.04\%}$
BCGP	(7.85 ± 1.12)e-4	**0**	$0.62 \pm 0.07\%$
HC-PINNs	(3.26 ± 0.65)e-3	**0**	$1.12 \pm 0.18\%$

B Experimental details

B.1 Experimental details for Wave 1D and equation Wave IBVP

These examples are concerned with the 1D wave equation

$$\partial_1^2 u - \partial_2^2 u = 0 \quad \text{in } \{x_2 > 0\}$$

with Dirichlet (Section 4.2) or Neumann (Section 5) boundary conditions:

$$\text{respectively} \quad u(x_1, 0) = 0 \quad \text{or} \quad \partial_2 u(x_1, 0) = 0.$$

Starting from EPGP, we calculate the characteristic variety $V = \{z \in \mathbb{C}^2 \colon z_1^2 - z_2^2 = 0\} = \{(z', \pm z') \colon z' \in \mathbb{C}\}$. We apply the formula B-EPGP: For $z' \in \mathbb{C}^{n-1} = \mathbb{C}$, we have $S_{z'} = \{(z', z'), (z', -z')\}$ and the restriction in B-EP gives

$$b(x_1, x_2; z') = \exp(z'(x_1 - x_2)) \mp \exp(z'(x_1 + x_2)) \quad \text{for } z' \in \mathbb{C},$$

where "$-$" corresponds to the Dirichlet case and "$+$" corresponds to the Neumann case. In our implementation we make the ad hoc restriction that $z' = \sqrt{-1}\omega$ is purely imaginary, leading to

$$\begin{aligned} b(x_1, x_2; z') &= \exp(\sqrt{-1}\omega(x_1 - x_2)) \mp \exp(\sqrt{-1}\omega(x_1 + x_2)) \\ &= \cos(\omega(x_1 - x_2)) + \sqrt{-1}\sin(\omega(x_1 - x_2)) \\ &\mp \cos(\omega(x_1 + x_2)) \mp \sqrt{-1}\sin(\omega(x_1 + x_2)). \end{aligned}$$

In the Dirichlet case "$-$" we get

$$b(x_1, x_2; z') = 2\sin(\omega x_1)\sin(\omega x_2) + 2\sqrt{-1}\cos(\omega x_1)\sin(\omega x_2).$$

After taking real and imaginary parts, we obtain the basis

$$b_1(x_1, x_2; \omega) = \sin(\omega x_1)\sin(\omega x_2), \quad b_2(x_1, x_2; \omega) = \cos(\omega x_1)\sin(\omega x_2) \quad \text{for } \omega \in \mathbb{R}.$$

In the Neumann case "$+$" we get

$$b(x_1, x_2; z') = 2\cos(\omega x_1)\cos(\omega x_2) + 2\sqrt{-1}\sin(\omega x_1)\cos(\omega x_2).$$

After taking real and imaginary parts, we obtain the basis

$$b_1(x_1, x_2; \omega) = \cos(\omega x_1)\cos(\omega x_2), \quad b_2(x_1, x_2; \omega) = \sin(\omega x_1)\cos(\omega x_2) \quad \text{for } \omega \in \mathbb{R}. \tag{3}$$

The case of Neumann boundary conditions is one of our central experiments, see Section 5. There, we compare B-EPGP with 6 probabilistic methods (CNO, FNO, WIGPR, EPGP, BCGP, HC-PINNs) and present the error to a ground truth in Table 2. Here we complement this experimental data with the PDE and BC error for all 7 methods in Table 4.

For reproducibility of our method, we list the following experimental details: We used $M = 1000$ basis functions 3 and between $N = 120$ or $N = 1,200$ data points in $\{0\} \times [0, 4]$. The training is performed in 3 steps, all of which using the standard Adam optimizer, with learning rate $0.1, 0.01, 0.001$, running for $20000, 20000, 1000$ epochs respectively. The training time was under 5 minutes for each experiment. To validate our results against the ground truths, we use the relative L_1 error metric, calculated at 161×81 points in $[0, 8] \times [0, 4]$. The best validation error was stored.

To ensure statistical significance, each experiment for Table 2 was repeated 10 times and we recorded the mean and standard deviation.

B.2 Experimental details for Table 1

We first give the B-EPGP bases for the six examples. The calculations are similar to the ones performed in Appendix B.1.

First we consider the 2D heat equation

$$\partial_1 u - \partial_2^2 u - \partial_3^2 u = 0 \quad \text{in } \{x_3 > 0\}, \tag{Heat 2D}$$

with Dirichlet boundary condition

$$u = 0 \quad \text{on } x_3 = 0. \tag{Dirichlet}$$

The B-EPGP basis is of the form

$$b_1(x; z) = \exp(-(z_2^2 + z_3^2)t)\cos(z_2 x_2)\sin(z_3 x_3),$$
$$b_2(x; z) = \exp(-(z_2^2 + z_3^2)t)\sin(z_2 x_2)\sin(z_3 x_3).$$

Also for equation Heat 2D but with Neumann boundary condition

$$\partial_3 u = 0 \quad \text{on } x_3 = 0, \tag{Neumann}$$

we get the B-EPGP basis

$$b_1(x; z) = \exp(-(z_2^2 + z_3^2)t)\cos(z_2 x_2)\cos(z_3 x_3),$$
$$b_2(x; z) = \exp(-(z_2^2 + z_3^2)t)\sin(z_2 x_2)\cos(z_3 x_3).$$

Next, for the 2D wave equation

$$\partial_1^2 u - \partial_2^2 u - \partial_3^2 u = 0 \quad \text{in } \{x_3 > 0\}, \tag{Wave 2D}$$

with Dirichlet boundary condition we have the B-EPGP basis

$$\begin{aligned} b_1(x; z) &= \cos(z_1 x_1 + z_2 x_2)\sin(z_3 x_3), \\ b_2(x; z) &= \sin(z_1 x_1 + z_2 x_2)\sin(z_3 x_3), \\ b_3(x; z) &= \cos(-z_1 x_1 + z_2 x_2)\sin(z_3 x_3), \\ b_4(x; z) &= \sin(-z_1 x_1 + z_2 x_2)\sin(z_3 x_3), \end{aligned} \qquad \text{for } z_1 = \sqrt{z_2^2 + z_3^2}. \tag{4}$$

For Wave 2D with Neumann boundary condition we have the B-EPGP basis

$$\begin{aligned} b_1(x; z) &= \cos(z_1 x_1 + z_2 x_2)\cos(z_3 x_3), \\ b_2(x; z) &= \sin(z_1 x_1 + z_2 x_2)\cos(z_3 x_3), \\ b_3(x; z) &= \cos(-z_1 x_1 + z_2 x_2)\cos(z_3 x_3), \\ b_4(x; z) &= \sin(-z_1 x_1 + z_2 x_2)\cos(z_3 x_3), \end{aligned} \qquad \text{for } z_1 = \sqrt{z_2^2 + z_3^2}.$$

Next for the 3D Laplace equation

$$\partial_1^2 u + \partial_2^2 u + \partial_3^2 u = 0 \quad \text{in } \{x_3 > 0\}, \tag{Laplace 3D}$$

with Dirichlet boundary condition we have the B-EPGP basis

$$\begin{aligned} b_1(x; z) &= \exp(z_1 x_1)\cos(z_2 x_2)\sin(z_3 x_3), \\ b_2(x; z) &= \exp(z_1 x_1)\sin(z_2 x_2)\sin(z_3 x_3), \\ b_3(x; z) &= \exp(-z_1 x_1)\cos(z_2 x_2)\sin(z_3 x_3), \\ b_4(x; z) &= \exp(-z_1 x_1)\sin(z_2 x_2)\sin(z_3 x_3), \end{aligned} \qquad \text{for } z_1 = \sqrt{z_2^2 + z_3^2}.$$

Finally for Laplace 3D with Neumann boundary condition we have the B-EPGP basis

$$\begin{aligned} b_1(x; z) &= \exp(z_1 x_1)\cos(z_2 x_2)\cos(z_3 x_3), \\ b_2(x; z) &= \exp(z_1 x_1)\sin(z_2 x_2)\cos(z_3 x_3), \\ b_3(x; z) &= \exp(-z_1 x_1)\cos(z_2 x_2)\cos(z_3 x_3), \\ b_4(x; z) &= \exp(-z_1 x_1)\sin(z_2 x_2)\cos(z_3 x_3), \end{aligned} \qquad \text{for } z_1 = \sqrt{z_2^2 + z_3^2}.$$

We give a few training details for the six experiments. Our predictors are Gaussian processes

$$p(x) = \sum_{j=1}^{M} w_j b(x; z_j)$$

where $M = 500$, w_j are normally distributed, and b are any of the b_i's listed above, corresponding to each experiment. The ground truths for each experiment are listed in Table 1. From these ground truths $u(x)$ we draw 2,000 data points $(X, u(X))$ with $X \in [0,1]^3$ on which we train the loss function (negative log marginal likelihood) described in Appendix C.3. The training is performed using the standard Adam optimizer with learning rate 0.002. This gives us the posterior distribution, whose expectation gives our approximant p. To validate our results against the ground truths, we use the relative RMSE metric, calculated at 5,000 points in $[0,1]^3$.

B.3 Experimental details for equation Wave 3D

For reproducibility of our method, we list the following experimental details concerning our implementation described in Section 5.1: We used $M = 500$ basis functions and $N = 41^3$ data points in $\{0\} \times [0,2]^3$. The training is performed in 3 steps, all of which are using the standard Adam optimizer, with learning rate $0.1, 0.01, 0.001$, running for $20000, 20000, 1000$ epochs respectively. The training time was under 40 minutes for each experiment. To validate our results against the ground truths, we use L1 error, calculated at 21^4 points in $[0,2]^4$. The best validation error was stored.

B.4 Experimental details for equation Helmholtz 3D

We recall equation Helmholtz 3D

$$\begin{cases} \partial_1^2 u + \partial_2^2 u + \partial_3^2 u + u = 0 & \text{in } [0,1]^3 \\ \partial_3 u + u = 0 & \text{on } x_3 = 0. \end{cases}$$

The EPGP basis is given by

$$\exp(z_1 x_1 + z_2 x_2 + z_3 x_3) \quad \text{where } z_1^2 + z_2^2 + z_3^2 + 1 = 0.$$

In particular, we can use the typical restriction in the study of acoustics, that $z = \sqrt{-1}\omega$ for $\omega \in \mathbb{R}^3$ to obtain $\omega_1^2 + \omega_2^2 + \omega_3^2 = 1$ and further parametrize $\omega = (\sin\theta\cos\varphi, \sin\theta\sin\varphi, \cos\theta)$ for $\theta \in [0,\pi]$, $\varphi \in [0, 2\pi]$. From the calculation of equation B-EP with coefficients $w^{\pm} \in \mathbb{C}$, we obtain

$$\begin{aligned} &w^+ \left(\sqrt{-1} z_3 \exp(\sqrt{-1} x'^{\top} \omega') + \exp(\sqrt{-1} x'^{\top} \omega')\right) \\ &+ w^- \left(-\sqrt{-1} z_3 \exp(\sqrt{-1} x'^{\top} \omega') + \exp(\sqrt{-1} x'^{\top} \omega')\right) = 0 \end{aligned}$$

By identifying real and imaginary parts of this equation we arrive at basis functions of two types

$$\begin{aligned} &b_1(x; z) = \cos(x_1\omega_1 + x_2\omega_2)(\omega_3 \cos(x_3\omega_3) - \sin(x_3\omega_3)), \\ &b_2(x; z) = \sin(x_1\omega_1 + x_2\omega_2)(\omega_3 \cos(x_3\omega_3) - \sin(x_3\omega_3)), \\ &\text{where } \omega_1 = \sin\theta\cos\varphi,\ \omega_2 = \sin\theta\sin\varphi,\ \omega_3 = \cos\theta \text{ for } \theta \in [0,\pi],\ \varphi \in [0, 2\pi]. \end{aligned}$$

This basis was used to obtain the results in Table 3. We used $M = 600$ basis functions and between 10-50 data points, scattered very sparsely in $[0,1]^3$. The training is performed using the standard Adam optimizer with learning rate 0.01, running for 15,000 epochs. The training time was under 1 minute for each experiment. To validate our results against the ground truths, we use the relative RMSE metric, calculated at 5,000 points in $[0,1]^3$. The best validation error was stored.

To limit test our method and ensure that the excellent results in Table 3 were not a result of a surprising coincidence, we also ran the experiment with only $N = 3$ data points. We reproduce the failing results in Table 5.

B.5 Experimental details for the numerical FTCS method

For the numerical method, we discretize the equation on the spatial domain $[0,1]^2$ using the explicit Forward Time Centered Space (FTCS) finite difference method. Let $U_{i,j}^n \approx u(t_n, x_i, y_j)$ denote the

Table 5: Failure of training of B-EPGP for the Helmholtz equation with Robin boundary condition if we sample only $N = 3$ data points.

Epoch	Train MSE	Current Val Rel RMSE
0	1.966005e+00	31.670%
3000	0.000000e+00	35.855%
6000	0.000000e+00	35.855%
9000	0.000000e+00	35.855%
12000	0.000000e+00	35.855%
14999	0.000000e+00	35.855%

numerical approximation at time $t_n = n\Delta t$ and spatial grid points $(x_i, y_j) = (i\Delta x, j\Delta y)$, assuming a uniform spatial step $\Delta x = \Delta y$. The time evolution is governed by the update rule:

$$U_{i,j}^{n+1} = U_{i,j}^n + \frac{D\Delta t}{\Delta x^2}\left(U_{i+1,j}^n + U_{i-1,j}^n + U_{i,j+1}^n + U_{i,j-1}^n - 4U_{i,j}^n\right).$$

The system is initialized using the exact free-space Gaussian profile, $U_{i,j}^0 = f(x_i, y_j)$, and strictly zero Dirichlet boundary conditions are enforced along the perimeter of the square for all $n \geq 0$. To guarantee numerical stability for this explicit 2D scheme, the temporal step size Δt is strictly bounded by the condition $\Delta t \leq \frac{\Delta x^2}{4D}$.

C Mathematical background and proofs

C.1 The Ehrenpreis–Palamodov theorem

We give a very precise version of the fact that *exponential-polynomial solutions are dense in the space of all solutions* for a linear PDE systems with constant coefficients. For notational clarity, we write $\mathbb{C}[\partial] := \mathbb{C}[\partial_1, \ldots, \partial_n]$, $x = (x_1, \ldots, x_n)$, and $z = (z_1, \ldots, z_n)$.

Theorem 2. *Let $\Omega \subset \mathbb{R}^n$ be an open convex set. Let $A = A(\partial) \in \mathbb{C}[\partial]^{\ell \times k}$ be an operator matrix of a system of linear PDEs with constant coefficients and $V = \{z \in \mathbb{C}^n \colon \ker A(z) \neq 0\}$ its characteristic variety. Then there exist a decomposition $V = \bigcup_{i=1}^m V_i$ into irreducible varieties V_i and a set of vector polynomials in $2n$ variables called* Noetherian multipliers *$\{p_{i,j}\}_{j=1\ldots r_i, i=1,\ldots m} \subset \mathbb{C}[x, z]^k$ such that solutions of the form*

$$\sum_{h=1}^{N}\sum_{i=1}^{m}\sum_{j=1}^{r_i} c_{i,j,h}\, p_{i,j}(x, z_{i,j,h})\, e^{x\cdot z_{i,j,h}} \quad \textit{with } z_{i,j,h} \in V_i \tag{EP}$$

are dense in the space of smooth solutions of $A(\partial)u(x) = 0$ in Ω.

We clarify the notion of smooth solution and the topology with respect to which we have density. First, we write $\mathcal{F} = C^\infty(\Omega)$ to be the space of smooth functions $\Omega \to \mathbb{C}$. This is a Fréchet space under the standard topology induced by the semi-norms $s_{a,b}(u) = \max_{|\alpha|=a, x\in\Omega_b} |\partial^\alpha u(x)|$, where $\Omega_b \uparrow \Omega$ is an increasing sequence of compact sets which exhausts Ω. This is to say that a sequence $u_q \to u$ in $\mathcal{F}$ if $s_{a,b}(u_q - u) \to 0$ for all a, b. Algebraically, $\mathcal{F}^k$ is a $\mathbb{C}[\partial]$-module under the action of differentiation.

Our solution space is then

$$\ker_{\mathcal{F}} A = \{u \in \mathcal{F}^k \colon A(\partial)u = 0\}.$$

The Ehrenpreis–Palamodov Theorem states that each element $u \in \ker_{\mathcal{F}} A$ can be approximated by $u_q \to u$ in $\mathcal{F}^k$ with solutions u_q of the form equation EP.

C.2 Computing the boundary characteristic variety and the B-EPGP basis

To compute the B-EPGP basis in general, we need to first be able to use the ansatz equation EP to solve linear PDEs in practice computation of the algebraic objects therein is crucial. To do this, we use the Macaulay2 command `solvepde` (Chen et al., 2023; Ait El Manssour et al., 2021) to obtain an analytic representation of the polynomials $p_{i,j}$ and varieties V_i in Theorem 2.

For every scalar operator considered in this paper, $A(\partial) = A_0(\partial') + \partial_n^2$ as in Theorem 1, and every boundary condition is $B(\partial) = \alpha\partial_n + \beta$ for constants $\alpha, \beta \in \mathbb{C}$, covering Dirichlet ($\alpha = 0, \beta = 1$), Neumann ($\alpha = 1, \beta = 0$), and Robin ($\alpha, \beta \neq 0$) conditions. In this case $S_{z'}$ is computed easily by hand: since $A(z', z_n) = A_0(z') + z_n^2$ has no linear term, $S_{z'} = \{(z', s), (z', -s)\}$ with $s = \sqrt{-A_0(z')}$, provided $A_0(z') \neq 0$ so that the two roots are distinct (the excluded locus $A_0(z') = 0$ has measure zero and does not affect the resulting prior). The single linear constraint $\sum_{z \in S_{z'}} B(z)w_z = 0$ then has the closed-form solution, unique up to scale,

$$w_{(z',s)} = \beta - \alpha s, \qquad w_{(z',-s)} = -(\beta + \alpha s),$$

giving

$$b(x; z') = e^{x'^\top z'} \big[2\beta \sinh(x_n s) - 2\alpha s \cosh(x_n s)\big].$$

For PDE systems, or multiple simultaneous boundary conditions, the general construction of $S_{z'}$ and the corresponding syzygy module requires Gröbner bases; we refer to Cox et al. (2015) and Lange-Hegermann (2018); Harkonen et al. (2023) for that general theory, which we only need for the systems in Appendix A.3.

In these more general cases, fix a splitting $z = (z', z_n)$ as in Section 4. The *boundary characteristic variety*

$$S_{z'} = \{(z', z_n) \in \mathbb{C}^n \colon A(z', z_n) = 0\}$$

is the fiber of V over z' under the projection $(z', z_n) \mapsto z'$. Choose dimensions so that $S_{z'}$ is generically finite, with cardinality equal to the degree of A in z_n (at most 2 for the operators we consider). We compute $S_{z'}$ as a Gröbner basis of $\langle A(z', z_n)\rangle \subset \mathbb{C}(z')[z_n]$ with respect to a lexicographic order eliminating z_n, which for our low-degree examples reduces to solving a univariate polynomial in z_n over the function field $\mathbb{C}(z')$.

Given $S_{z'}$, the coefficients w_z in equation B-EP must satisfy linear relations $\sum_{z \in S_{z'}} B(z)w_z = 0$. They are given by the *syzygy module* (i.e. the generating set of a kernel) over $\mathbb{C}[z']$; a basis of it (computable via Gröbner bases, or by direct linear algebra when $S_{z'}$ is small) yields a basis of the B-EPGP feature map $b(\,\cdot\,; z')$ in equation B-EP. Since $|S_{z'}| \leq 2$ throughout our experiments, this reduces to solving one linear equation in two unknowns by hand.

All of the above is carried out once, symbolically, before training: for every example in this paper the full computation (characteristic variety, boundary variety, and syzygy basis) completes in fractions of one second and contributes nothing to per-iteration training cost. Even bigger examples tend to terminate instantaneously.

C.3 Gaussian process regression for B-EPGP

We express our predictions in the equation B-EPGP basis

$$\phi(x) = \sum_{j=1}^{M} w_j b(x; z_j), \tag{5}$$

where $b(\,\cdot\,, ; z)$ are precomputed symbolically according to Section C.2.

We assume that our data is given by $y_h \approx \phi(x_h)$ for $h = 1, \ldots, N$. We write $W = (w_j) \in \mathbb{R}^M$, $Z = (z_j)_{j=1}^M$ restricted according to Appendix C.2, $X = (x_h) \in \mathbb{R}^{N \times n}$, $Y = (y_h) \in \mathbb{R}^N$.

We model the weights as $W \sim \mathcal{N}(0, \frac{1}{M}\Sigma)$ with $\Sigma = \operatorname{diag}(\sigma_j^2)_{j=1}^M$, and assume Gaussian noise $Y - \phi(X) \sim \mathcal{N}(0, \sigma_0^2 I_N)$. Writing $\Phi = (b(x_h; z_j)) \in \mathbb{R}^{N \times M}$, we have $\phi(X) = \Phi W$, so $\phi(X) \sim \mathcal{N}(0, \frac{1}{M}\Phi\Sigma\Phi^\top)$. We write $\sigma^2 = (\sigma_j^2)_{j=0}^M$ for the vector of variance parameters. By the Woodbury identity and the matrix determinant lemma, as is standard for weight-space Gaussian processes with finitely many features (Lázaro-Gredilla et al., 2010; Harkonen et al., 2023), the marginal log likelihood of this model is maximized if the function

$$L(Z, \sigma^2, \sigma_0^2; X, Y) = \frac{1}{\sigma_0^2}(\|Y\|^2 - Y^\top \Phi G^{-1} \Phi^\top Y) + (N - M)\log\sigma_0^2 + \log\det\Sigma + \log\det G$$

is minimized, where $G = M\sigma_0^2\Sigma^{-1} + \Phi^\top\Phi$; L equals twice the negative log marginal likelihood up to the additive constant $N \log 2\pi - M \log M$. We then use stochastic gradient descent (Adam

optimizer) to minimize $L(Z, \sigma^2)$ for (Z_*, σ_*^2) and the explicit formula for $W_* = G^{-1}\Phi^\top Y$ which we plug in equation 5, giving our final prediction.

We stress that the B-EPGP basis is computed analytically (either by hand or by computer algebra, see Appendix C.2) *once* before training even starts. It therefore does not affect training time.

C.4 Proof of Theorem 1

Recall the notation $x = (x', x_n)$ where $x' = (x_1, x_2, \ldots, x_{n-1})$. We repeat the statement of our main new theoretical result:

Theorem 3. *Suppose that $A(\partial) = A_0(\partial') + \partial_n^2$ and that $B(\partial)u = u$ or $B(\partial)u = \partial_n u$ in BVP. Then any solution $u \in C^\infty(\mathbb{R}^{n-1} \times [0, \infty))$ of BVP can be approximated in the Fréchet topology by functions of the form B-EPGP.*

Proof. First consider the case of $B(\partial) = 1$. Let u be as in the statement of the theorem. Let

$$v(x', x_n) = \begin{cases} u(x', x_n) & x_n \geq 0 \\ -u(x', -x_n) & x_n < 0. \end{cases}$$

We claim that $A(\partial)v = 0$ in $\mathbb{R}^n$ in the distributional sense. To see this, let $\varphi \in C_c^\infty(\mathbb{R}^n)$ and denote A_0^* the adjoint operator of A_0. We calculate

$$\begin{aligned}
&\int_{\mathbb{R}^n} v(A_0^*(\partial')\varphi + \partial_n^2\varphi)\,\mathrm{d}x \\
&\quad= \int_{\mathbb{R}^{n-1}}\int_0^\infty u(x', x_n)(A_0^*(\partial')\varphi(x) + \partial_n^2\varphi(x))\,\mathrm{d}x_n\,\mathrm{d}x' \\
&\quad- \int_{\mathbb{R}^{n-1}}\int_{-\infty}^0 u(x', -x_n)(A_0^*(\partial')\varphi(x) + \partial_n^2\varphi(x))\,\mathrm{d}x_n\,\mathrm{d}x' \\
&\quad= \int_{\mathbb{R}^{n-1}}\int_0^\infty u(x', x_n)(A_0^*(\partial')\varphi(x) + \partial_n^2\varphi(x))\,\mathrm{d}x_n\,\mathrm{d}x' \\
&\quad- \int_{\mathbb{R}^{n-1}}\int_0^\infty u(x)(A_0^*(\partial')\varphi(x', -x_n) + \partial_n^2\varphi(x', -x_n))\,\mathrm{d}x_n\,\mathrm{d}x' \\
&\quad= \int_{\mathbb{R}^{n-1}}\int_0^\infty A_0(\partial')u(x', x_n)\varphi(x)\,\mathrm{d}x - \int_{\mathbb{R}^{n-1}} u(x', 0)\partial_n\varphi(x', 0)\,\mathrm{d}x' \\
&\quad- \int_{\mathbb{R}^{n-1}}\int_0^\infty \partial_n u(x', x_n)\partial_n\varphi(x)\,\mathrm{d}x_n\,\mathrm{d}x' \\
&\quad- \int_{\mathbb{R}^{n-1}}\int_0^\infty A_0(\partial')u(x)\varphi(x', -x_n)\,\mathrm{d}x_n\,\mathrm{d}x' - \int_{\mathbb{R}^{n-1}} u(x', 0)\partial_n\varphi(x', 0)\,\mathrm{d}x' \\
&\quad- \int_{\mathbb{R}^{n-1}}\int_0^\infty \partial_n u(x)\partial_n\varphi(x', -x_n)\,\mathrm{d}x_n\,\mathrm{d}x' \\
&\quad= \int_{\mathbb{R}^{n-1}}\int_0^\infty A(\partial)u\varphi\,\mathrm{d}x + \int_{\mathbb{R}^{n-1}} \partial_n u(x', 0)\varphi(x', 0)\,\mathrm{d}x' \\
&\quad- \int_{\mathbb{R}^{n-1}}\int_0^\infty A(\partial)u(x)\varphi(x', -x_n)\,\mathrm{d}x'\,\mathrm{d}x_n - \int_{\mathbb{R}^{n-1}} \partial_n u(x', 0)\varphi(x', 0)\,\mathrm{d}x',
\end{aligned}$$

which equals zero so indeed v satisfies the PDE. Moreover, v is in $C^\infty(\mathbb{R}^n)$: from $\partial_n^2 u = -A_0(\partial')u$ we get $\partial_n^{2k}u(x', 0) = (-A_0(\partial'))^k u(x', 0) = 0$ for all k, so all derivatives of the odd extension match across $\{x_n = 0\}$.

Consider an Ehrenpreis–Palamodov approximation w of v, so $A(\partial)w = 0$ and w has the form

$$w(x) = \sum_{j=1}^M w_j \exp(x^\top z_j) \quad \text{where } A(z_j) = 0.$$

Let

$$\tilde{w}(x) = \frac{1}{2}\left(w(x', x_n) - w(x', -x_n)\right)$$

which has the following properties:

First, $\tilde{w}$ is generated by the B-EPGP basis

$$b(x; z_j) = \exp(x^\top z_j) - \exp(x^\top (z_j', -z_n)) \quad \text{for } A(z_j) = 0.$$

and therefore it satisfies both boundary condition and PDE exactly.

Second, v can be approximated by $\tilde{w}$ because v is odd in x_n, so it satisfies

$$v(x) = \frac{1}{2} \left(v(x', x_n) - v(x', -x_n) \right).$$

Therefore $u = v$ in $\{x \in \mathbb{R}^n \colon x_n \geq 0\}$ can be approximated with $\tilde{w}$ in the halfspace.

The proof of this case is complete. In the case of the Neumann boundary condition we replace the odd extension in the definition of v above by the even extension

$$v(x', x_n) = \begin{cases} u(x', x_n) & x_n \geq 0 \\ u(x', -x_n) & x_n < 0. \end{cases}$$

The proof can then be completed along the same lines. □

C.5 Applications to control

Physics informed machine learning models play an important role in ODE theory and control. Geist & Trimpe (2020) construct GP models for control for rigid body dynamics. Besginow & Lange-Hegermann (2022) uses the special case of Ehrenpreis-Palamodov for ODEs to construct such models for general linear ODE systems with constant coefficients, leading to control approaches (Besginow et al., 2025; Tebbe et al., 2024).