# Solving and Learning Nonlinear PDEs with Gaussian Processes

Yifan Chen, Caltech

Joint work with

Bamdad Hosseini, Houman Owhadi, Florian Schaefer and Andrew Stuart

Rough Path Interest Group, 2022

# Contents

# Roadmap

# Numerical Approximation and Inference

- **Partial Differential Equations**: infinite degrees of freedom (DOF)

$$\mathcal{F}(x, t, u, \partial_t u, \nabla_x u, \nabla_x^2 u, a, \xi, ...) = 0$$

  - Stationary PDEs, time dependent, inverse problems, UQ, ...

- **Numerical Approximation** (finite DOF) designed by experts
  - Finite difference/element/volume
  - Spectral methods
  - Boundary integral methods
  - Meshless methods, collocation methods
  - Multiscale methods, numerical homogenization, ...

- **Inference and ML** to automate the finite $\leftrightarrow$ infinite DOF process
  - Gaussian process (GP) and kernel methods for numerical integration
  - GPs and kernel methods for ODEs, linear PDEs
  - Bayes probabilistic numerics, Bayes numerical analysis, UQ
  - Physics informed ML (Deep Ritz methods, PINNs, SDEs...)
  - Operator learning (Kernels, Neural Operators, DeepONets), ...

# This talk

**Our Goal**

A general GP framework for solving and learning nonlinear PDEs

- Intepretable, convergent and amenable to numerical analysis[1]
  - generalize RBF collocation methods and meshless kernel methods
- Near-linear time and space complexity implementation[2]
  - quantitative screening effects for GPs with PDE measurements
- Hierarchical parameter learning in the GP, or kernel learning[3]
  - consistency analysis of Kernel Flow and Emperical Bayes

[1] Yifan Chen, Bamdad Hosseini, Houman Owhadi, and Andrew M Stuart. "Solving and learning nonlinear pdes with gaussian processes". In: *Journal of Computational Physics* (2021).

[2] Yifan Chen, Florian Schaefer, and Houman Owhadi. "Sparse Cholesky Factorization for Solving Nonlinear PDEs via Gaussian Processes". In preparation.

[3] Yifan Chen, Houman Owhadi, and Andrew Stuart. "Consistency of empirical Bayes and kernel flow for hierarchical parameter estimation". In: *Mathematics of Computation* (2021).

# Roadmap

# A Nonlinear Elliptic PDE Example

- Consider the stationary elliptic PDE

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

  - Domain $\Omega \subset \mathbb{R}^d$.
  - PDE data $f, g : \Omega \to \mathbb{R}$.

- PDE has a unique strong/classical solution $u^\star$.

# A Nonlinear Elliptic PDE: The Methodology

1. Choose a kernel $K : \overline{\Omega} \times \overline{\Omega} \to \mathbb{R}$
   - Corresponding RKHS $\mathcal{U}$ with norm $\|\cdot\|$

2. Choose some collocation points
   - $X^{\mathsf{int}} = \{\mathbf{x}_1^{\mathsf{int}}, \ldots, \mathbf{x}_{M^{\mathsf{int}}}^{\mathsf{int}}\} \subset \Omega$
   - $X^{\mathsf{bd}} = \{\mathbf{x}_1^{\mathsf{bd}}, \ldots, \mathbf{x}_{M^{\mathsf{bd}}}^{\mathsf{bd}}\} \subset \partial\Omega$

3. Solve the optimization problem

$$\begin{cases} \underset{u \in \mathcal{U}}{\text{minimize}} \ \|u\| \\ \text{s.t.} \quad -\Delta u(\mathbf{x}_m) + \tau(u(\mathbf{x}_m)) = f(\mathbf{x}_m), \quad \text{for } \mathbf{x}_m \subset X^{\mathsf{int}} \\ \qquad\qquad\qquad\qquad\qquad\quad u(\mathbf{x}_n) = g(\mathbf{x}_n), \qquad \text{for } \mathbf{x}_n \subset X^{\mathsf{bd}} \end{cases}$$

# Bayes Inference Interpratation of the Methodology

1. Choose a kernel $K : \overline{\Omega} \times \overline{\Omega} \to \mathbb{R}$ (Choose the prior $\mathcal{GP}(0, K)$)
   - Corresponding RKHS $\mathcal{U}$ with norm $\|\cdot\|$
2. Choose some collocation points (Choose the data/likelihood)
   - $X^{\mathsf{int}} = \{\mathbf{x}_1^{\mathsf{int}}, \ldots, \mathbf{x}_{M^{\mathsf{int}}}^{\mathsf{int}}\} \subset \Omega$
   - $X^{\mathsf{bd}} = \{\mathbf{x}_1^{\mathsf{bd}}, \ldots, \mathbf{x}_{M^{\mathsf{bd}}}^{\mathsf{bd}}\} \subset \partial\Omega$
3. Solve the optimization problem (Find the "MAP")

$$\begin{cases} \underset{u \in \mathcal{U}}{\text{minimize}} \ \|u\| \\ \text{s.t.} \quad -\Delta u(\mathbf{x}_m) + \tau(u(\mathbf{x}_m)) = f(\mathbf{x}_m), \quad \text{for } \mathbf{x}_m \subset X^{\mathsf{int}} \\ \qquad\qquad\qquad\qquad u(\mathbf{x}_n) = g(\mathbf{x}_n), \qquad \text{for } \mathbf{x}_n \subset X^{\mathsf{bd}} \end{cases}$$

Generalize linear PDEs in Bayes probabilistic numerical methods[4][5]

- Solving PDEs as a Bayes inverse problem

[4] Houman Owhadi. "Bayesian numerical homogenization". In: *Multiscale Modeling & Simulation* 13.3 (2015), pp. 812–828.

[5] Jon Cockayne, Chris J Oates, Timothy John Sullivan, and Mark Girolami. "Bayesian probabilistic numerical methods". In: *SIAM Review* 61.4 (2019), pp. 756–789.

# Roadmap

# Introducing Slack Variables

$$
\begin{cases}
\underset{u \in \mathcal{U}}{\text{minimize}} \ \|u\| \\
\text{s.t.} \quad -\Delta u(\mathbf{x}_m) + u(\mathbf{x}_m)^3 = f(\mathbf{x}_m), \quad \text{for } \mathbf{x}_m \subset X^{\mathsf{int}} \\
\qquad\qquad\qquad\qquad u(\mathbf{x}_n) = g(\mathbf{x}_n), \quad \text{for } \mathbf{x}_n \subset X^{\mathsf{bd}}
\end{cases}
$$

$$
\Updownarrow \ (N = M^{\mathsf{bd}} + 2M^{\mathsf{int}})
$$

$$
\begin{cases}
\underset{\mathbf{z}=(\mathbf{z}^{\mathsf{bd}},\mathbf{z}^{\mathsf{int}},\mathbf{z}_{\Delta}^{\mathsf{int}})\in\mathbb{R}^N}{\text{minimize}} \begin{cases}
\underset{u \in \mathcal{U}}{\text{minimize}} \ \|u\| \\
\text{s.t.} \quad u(X^{\mathsf{bd}}) = \mathbf{z}^{\mathsf{bd}} \\
\qquad u(X^{\mathsf{int}}) = \mathbf{z}^{\mathsf{int}} \\
\qquad \Delta u(X^{\mathsf{int}}) = \mathbf{z}_{\Delta}^{\mathsf{int}}
\end{cases} \\
\text{s.t.} \quad -\mathbf{z}^{\mathsf{int}} + \tau(\mathbf{z}_{\Delta}^{\mathsf{int}}) = f(X^{\mathsf{int}}) \\
\qquad \mathbf{z}^{\mathsf{bd}} = g(X^{\mathsf{bd}})
\end{cases}
$$

# Inner optimization

- The inner problem is linear

$$\begin{aligned} &\underset{u\in\mathcal{U}}{\text{minimize}} \ \ \|u\| \\ &\text{s.t.} \quad u(X^{\mathsf{bd}}) = \mathbf{z}^{\mathsf{bd}}, u(X^{\mathsf{int}}) = \mathbf{z}^{\mathsf{int}}, \Delta u(X^{\mathsf{int}}) = \mathbf{z}^{\mathsf{int}}_{\Delta} \end{aligned}$$

- Measurement vector $\boldsymbol{\phi} := (\delta_{X^{\mathsf{bd}}}, \delta_{X^{\mathsf{int}}}, \delta_{X^{\mathsf{int}}} \circ \Delta) \in (\mathcal{U}^*)^{\otimes N}$
- Kernel vector and matrix

$$K(\mathbf{x},\boldsymbol{\phi}) = \left(K(\mathbf{x}, X^{\mathsf{bd}}), K(\mathbf{x}, X^{\mathsf{int}}), \Delta_{\mathbf{y}} K(\mathbf{x}, X^{\mathsf{int}})\right) \in \mathbb{R}^N$$

$$K(\boldsymbol{\phi},\boldsymbol{\phi}) = \begin{pmatrix} K(X^{\mathsf{bd}}, X^{\mathsf{bd}}) & K(X^{\mathsf{bd}}, X^{\mathsf{int}}) & \Delta_{\mathbf{y}} K(X^{\mathsf{bd}}, X^{\mathsf{int}}) \\ K(X^{\mathsf{int}}, X^{\mathsf{bd}}) & K(X^{\mathsf{int}}, X^{\mathsf{int}}) & \Delta_{\mathbf{y}} K(X^{\mathsf{int}}, X^{\mathsf{int}}) \\ \Delta_{\mathbf{x}} K(X^{\mathsf{int}}, X^{\mathsf{bd}}) & \Delta_{\mathbf{x}} K(X^{\mathsf{int}}, X^{\mathsf{int}}) & \Delta_{\mathbf{x}}\Delta_{\mathbf{y}} K(X^{\mathsf{int}}, X^{\mathsf{int}}) \end{pmatrix} \in \mathbb{R}^{N\times N}$$

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

# Representation of the Minimizer

Combine the two level optimization:

**Representer theorem**

Every minimizer $u^\dagger$ can be represented as

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

where the vector $\mathbf{z}^\dagger \in \mathbb{R}^N$ is a minimizer of

$$\begin{cases} \min\limits_{\mathbf{z}\in\mathbb{R}^N} & \mathbf{z}^T K(\boldsymbol{\phi}, \boldsymbol{\phi})^{-1}\mathbf{z} \\ \text{s.t.} & F(\mathbf{z}) = \mathbf{y} \end{cases}$$

- Function $F : \mathbb{R}^N \to \mathbb{R}^M$ depends on PDE collocation constraints
- $\mathbf{y}$ contains PDE boundary and RHS data

# Roadmap

# Towards A Practical Algorithm

Quadratic optimization with nonlinear constraints

- A simple linearization algorithm $\mathbf{z}^k \to \mathbf{z}^{k+1}$

$$\begin{cases} \min\limits_{\mathbf{z}\in\mathbb{R}^N} & \mathbf{z}^T K(\boldsymbol{\phi},\boldsymbol{\phi})^{-1}\mathbf{z} \\ \text{s.t.} & F(\mathbf{z}^k) + F'(\mathbf{z}^k)(\mathbf{z}-\mathbf{z}^k) = \mathbf{y}. \end{cases}$$

"Newton's iteration for the nonlinear PDE"

- Poor conditioning of $K(\boldsymbol{\phi},\boldsymbol{\phi})$, and scale imbalance between blocks
  Adding scale-aware regularization $K(\boldsymbol{\phi},\boldsymbol{\phi}) + \lambda\text{diag}(K(\boldsymbol{\phi},\boldsymbol{\phi}))$

# Roadmap

# Numerical Experiments: Stationary Problems

- Nonlinear Elliptic Equation, $\tau(u) = u^3$

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

- Truth: $d = 2$, $u^\star(\mathbf{x}) = \sin(\pi x_1)\sin(\pi x_2) + 4\sin(4\pi x_1)\sin(4\pi x_2)$
- Kernel: $K(\mathbf{x}, \mathbf{y}; \sigma) = \exp(-\frac{|\mathbf{x}-\mathbf{y}|^2}{2\sigma^2})$

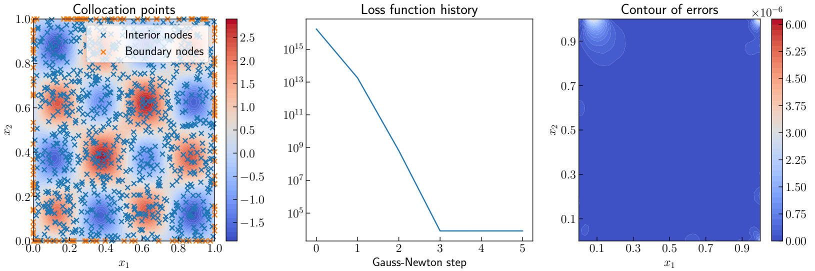



Figure: $N_{\text{domain}} = 900, N_{\text{boundary}} = 124$

# Convergence Study

- For $\tau(u) = 0, u^3$, use Gaussian kernel with lengthscale $\sigma$
- $L^2, L^\infty$ accuracy, compared with Finite Difference (FD)

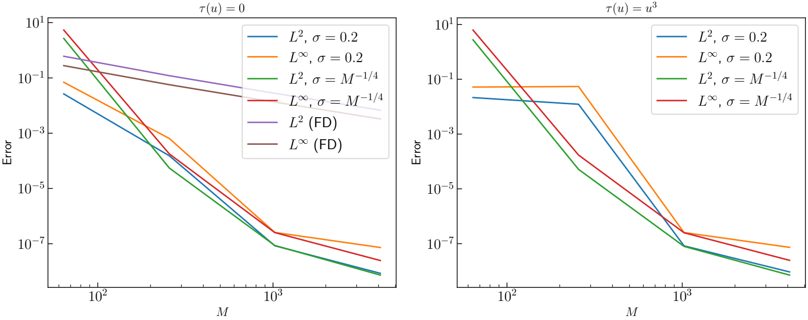
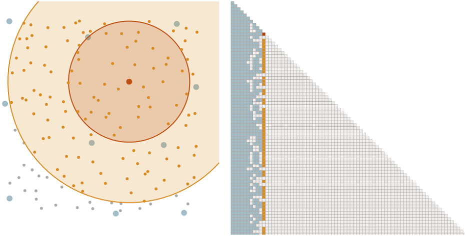



Figure: Convergence of the kernel method is fast, since the solution is smooth

# Roadmap

# Scalability: Taming the Dense Kernel Matrice

Sparse Cholesky factor for kernel matrices under coarse to fine ordering[6]



Coarse to fine: max-min ordering

$$x_k = \mathrm{argmax}_{x_i} d(x_i, \{x_j, 1 \le j < k\})$$

with lengthscale $l_k = d(x_k, \{x_j, 1 \le j < k\})$

[6] F Schäfer, TJ Sullivan, and H Owhadi. "Compression, inversion, and approximate PCA of dense kernel matrices at near-linear computational complexity". In: *Multiscale Modeling & Simulation* 19.2 (2021), pp. 688–730.

# Why Sparse? Cholesky Factors and Screening Effects

Let $\Theta \in \mathbb{R}^{d\times d}$, $\Theta_{ij} = k(x_i, x_j)$, and $X \sim \mathcal{N}(0, \Theta)$

- Cholesky factor of the covariance matrix $\Theta = LL^T$

$$\frac{L_{ij}}{L_{jj}} = \frac{\mathrm{Cov}[X_i, X_j | X_{1:j-1}]}{\mathrm{Var}[X_j | X_{1:j-1}]} \qquad (i \geq j)$$

- Cholesky factor of the precision matrix $\Theta^{-1} = UU^T$

$$\frac{U_{ij}}{U_{jj}} = (-1)^{i\neq j} \frac{\mathrm{Cov}[X_i, X_j | X_{1:j-1\setminus\{i\}}]}{\mathrm{Var}[X_j | X_{1:j-1\setminus\{i\}}]} \qquad (i \leq j)$$

Screening effects: $x_{1:j}$ ordered from coarse to fine; scale of $x_j$ is $l_j$, then for certain kernel arsing from PDEs [8]

$$\mathrm{Cov}[X_i, X_j | X_{1:j-1}] \lesssim \exp\left(-\frac{d(x_i, x_j)}{l_j}\right)$$

[7] Michael L Stein. "The screening effect in kriging". In: *Annals of statistics* 30.1 (2002), pp. 298–323.

[8] Schäfer, Sullivan, and Owhadi, "Compression, inversion, and approximate PCA of dense kernel matrices at near-linear computational complexity".

# Screening Effects with PDE measurements

Recall the kernel matrices

$$\begin{pmatrix} K(X^{\text{bd}}, X^{\text{bd}}) & K(X^{\text{bd}}, X^{\text{int}}) & \Delta_{\mathbf{y}} K(X^{\text{bd}}, X^{\text{int}}) \\ K(X^{\text{int}}, X^{\text{bd}}) & K(X^{\text{int}}, X^{\text{int}}) & \Delta_{\mathbf{y}} K(X^{\text{int}}, X^{\text{int}}) \\ \Delta_{\mathbf{x}} K(X^{\text{int}}, X^{\text{bd}}) & \Delta_{\mathbf{x}} K(X^{\text{int}}, X^{\text{int}}) & \Delta_{\mathbf{x}} \Delta_{\mathbf{y}} K(X^{\text{int}}, X^{\text{int}}) \end{pmatrix}$$

How to order when there are derivative measurements?

- Order pointwise measurements from coarse to fine
- PDE measurements follow behind (with the same ordering)

**Theorem: screening effects hold for such ordering**

Theory: need technical assumptions

- The kernel is the Green function of some differential operator $\mathcal{L} : H_0^s(\Omega) \to H^{-s}(\Omega)$

Practice: works more generally

# Near Linear Complexity by Sparse Cholesky

- Ignore correlation beyond $d(x, x_j) \geq \rho l_j$ (which is $O(\exp(-\rho))$)
- Once ordering and sparsity pattern determined, use KL minimization algorithm[9]: $O(N\rho^d)$ memory and $O(N\rho^{2d})$ time

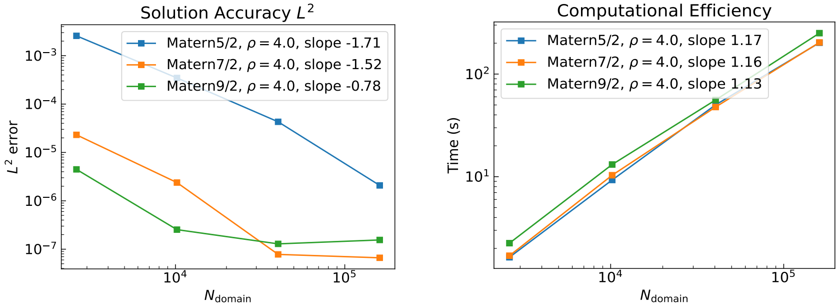



Figure: Run 3 GN iterations. Accuracy floor due to finite $\rho$ and regularization

[9] Florian Schäfer, Matthias Katzfuss, and Houman Owhadi. "Sparse Cholesky Factorization by Kullback–Leibler Minimization". In: *SIAM Journal on Scientific Computing* 43.3 (2021), A2019–A2046.

# Roadmap

# Numerical Experiments: Time Dependent Problems

Viscous Burgers' Equation

- Viscosity $\nu = 0.02$

$$\begin{cases} \partial_t u + u\partial_s u - \nu\partial_s^2 u = 0, \qquad \forall (s,t) \in (-1,1) \times (0,1]. \\ u(s,0) = -\sin(\pi s), \\ u(-1,t) = u(1,t) = 0. \end{cases}$$

- Shock when $\nu = 0$. Problem harder for smaller $\nu$
- Choose an anisotropic spatio-temperal GP

# Numerical Experiments: Viscous Burgers' Equation

- Kernel: $K((s,t),(s',t')) = \exp\left(-20^2|s-s'|^2 - 3^2|t-t'|^2\right)$

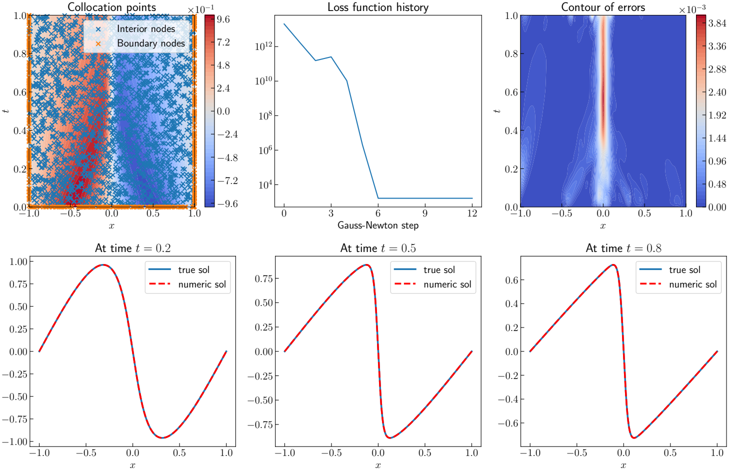



Figure: $N_{\text{domain}} = 2000, N_{\text{boundary}} = 400$

# Push to Small Viscosity

Discretize in time first, then apply the methodology to the resulting spatial PDE: dimension of kernel matrices is reduced

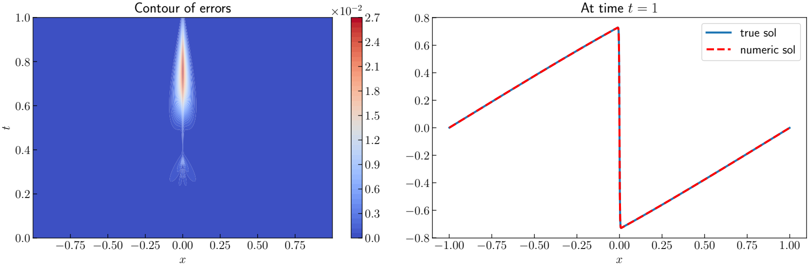



Figure: $\nu = 10^{-3}$; number of spatial points $2000$; time step size $0.01$; Matern7/2 kernel with lengthscale 0.02; use $2$ GN iterations

At time $t = 1$, $L^2$ accuracy: $10^{-4}$

- Observation: accuracy not monotone regarding time $t$
- Implication: further improvement through time-adaptive kernels

# Roadmap

# Numerical Experiments: Inverse Problems

Darcy Flow inverse problems

$$\begin{cases} \min_{u,a} \|u\|_K^2 + \|a\|_\Gamma^2 + \frac{1}{\gamma^2}\sum_{j=1}^{I}|u(\mathbf{x}_j)-o_j|^2, \\ \text{s.t.} \quad -\mathsf{div}(\exp(a)\nabla u)(\mathbf{x}_m) = 1, \quad \forall \mathbf{x}_m \in (0,1)^2 \\ \qquad\qquad u(\mathbf{x}_m) = 0, \quad \forall \mathbf{x}_m \in \partial(0,1)^2. \end{cases}$$

- Recover $a$ from pointwise measurements of $u$
- Model $(u,a)$ as independent GPs
- Impose PDE constraints and formulate Bayesian inverse problem

# Numerical Experiments: Darcy Flow

- Kernel $K(\mathbf{x}, \mathbf{x}'; \sigma) = \exp\left(-\frac{|\mathbf{x}-\mathbf{x}'|^2}{2\sigma^2}\right)$ for both $u$ and $a$

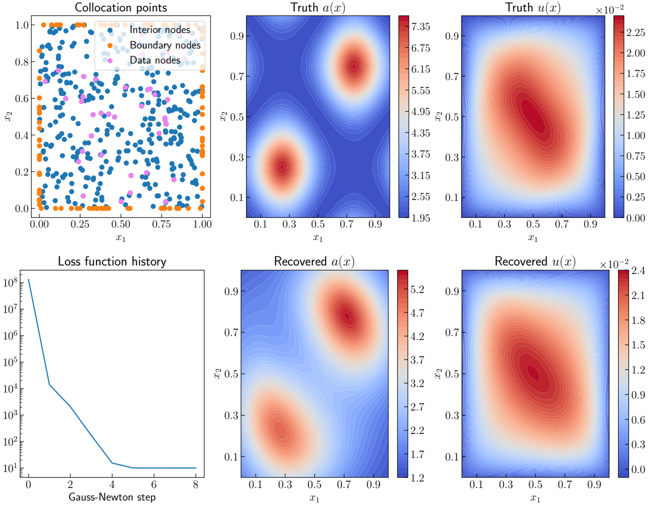



Figure: $N_{\text{domain}} = 400, N_{\text{boundary}} = 100, N_{\text{observation}} = 50$

# Roadmap

# Theoretical Foundation: Consistency

Consistency of the minimizer

$$\begin{cases} \min\limits_{u \in \mathcal{U}} & \|u\| \\ \text{s.t.} & \text{PDE constraints at } \{\mathbf{x}_1, \ldots, \mathbf{x}_M\} \in \overline{\Omega}. \end{cases}$$

### Convergence theory

- $K$ is chosen so that
  - $\mathcal{U} \subseteq H^s(\Omega)$ for some $s > s^*$ where $s^* = d/2 +$ order of PDE.
  - $u^\star \in \mathcal{U}$.
- Fill distance of $\{\mathbf{x}_1, \ldots, \mathbf{x}_M\} \to 0$ as $M \to \infty$.

Then as $M \to \infty$, $u^\dagger \to u^\star$ pointwise in $\Omega$ and in $H^t(\Omega)$ for $t \in (s^*, s)$.

# Roadmap

# Theoretical Foundation: Kernel Learning

Hierarchical parameters in the kernel $K_\theta$

- Good $\theta$ improves the performance

Algorithms for learning $\theta$: another level of optimization

- Bayes approach built in GPs: e.g. Empirical Bayes (EB)
- Kernel Flow (KF)[10]: a variant of cross-validation

$$\min_\theta \ \mathbb{E}_\pi \frac{\|u^\dagger(\cdot, X, \theta) - u^\dagger(\cdot, \pi X, \theta)\|_{K_\theta}^2}{\|u^\dagger(\cdot, X, \theta)\|_{K_\theta}^2}$$

- $u^\dagger(\cdot, X, \theta)$ is the solution using collocation points $X$ and kernel $K_\theta$
- $\pi X$ is a subsampling of $X$
- $\|\cdot\|_{K_\theta}$ is the RKHS norm for the kernel $K_\theta$; attain explicit formula due to representer theorem

Our result: Consistency of learning regularity of Matérn-like kernels

- EB and KF learn different parameters for linear problems

[10] Houman Owhadi and Gene Ryan Yoo. "Kernel flows: From learning kernels from data into the abyss". In: *Journal of Computational Physics* 389 (2019), pp. 22–47.

# Roadmap

# Take-aways

Solving and Learning Nonlinear PDEs with Gaussian Processes

Algorithm

- A simple framework for solving and learning nonlinear PDEs
- Near-linear complexity treatment of the dense kernel matrices
- Experiments: stationary PDEs, time dependent, inverse problems
- Future work: parametric PDEs, high dimensional PDEs, UQ, ...

Convergence theory

- Consistency as fill-in distance goes to 0 (asymptotic only)
- Future work: convergence rates?

Kernel learning (The hard part)

- Consistency of Kernel Flow and Empirical Bayes for linear problem :)

# Thank you!

# References


Chen, Yifan, Bamdad Hosseini, Houman Owhadi, and Andrew M Stuart. "Solving and learning nonlinear pdes with gaussian processes". In: *Journal of Computational Physics* (2021).

Chen, Yifan, Florian Schaefer, and Houman Owhadi. "Sparse Cholesky Factorization for Solving Nonlinear PDEs via Gaussian Processes". In preparation.

Chen, Yifan, Houman Owhadi, and Andrew Stuart. "Consistency of empirical Bayes and kernel flow for hierarchical parameter estimation". In: *Mathematics of Computation* (2021).

Owhadi, Houman. "Bayesian numerical homogenization". In: *Multiscale Modeling & Simulation* 13.3 (2015), pp. 812–828.

Cockayne, Jon, Chris J Oates, Timothy John Sullivan, and Mark Girolami. "Bayesian probabilistic numerical methods". In: *SIAM Review* 61.4 (2019), pp. 756–789.

Schäfer, F, TJ Sullivan, and H Owhadi. "Compression, inversion, and approximate PCA of dense kernel matrices at near-linear computational complexity". In: *Multiscale Modeling & Simulation* 19.2 (2021), pp. 688–730.

Stein, Michael L. "The screening effect in kriging". In: *Annals of statistics* 30.1 (2002), pp. 298–323.

Schäfer, Florian, Matthias Katzfuss, and Houman Owhadi. "Sparse Cholesky Factorization by Kullback–Leibler Minimization". In: *SIAM Journal on Scientific Computing* 43.3 (2021), A2019–A2046.

Owhadi, Houman and Gene Ryan Yoo. "Kernel flows: From learning kernels from data into the abyss". In: *Journal of Computational Physics* 389 (2019), pp. 22–47.