Probabilistic Meshless Methods for Bayesian Inverse Problems

Jon Cockayne

December 15, 2016

Co-Authors

Chris Oates

Tim Sullivan

Mark Girolami

What is PN?

Many problems in mathematics have no analytical solution, and must be solved numerically.

In Probabilistic Numerics we phrase such problems as inference problems and construct a probabilistic description of this error.

What is PN?

Many problems in mathematics have no analytical solution, and must be solved numerically.

In Probabilistic Numerics we phrase such problems as inference problems and construct a probabilistic description of this error.

This is not a new idea[1]!

Lots of recent development on Integration, Optimization, ODEs, PDEs... see `http://probnum.org/`

[2][Kadane, 1985, Diaconis, 1988, O'Hagan, 1992, Skilling, 1991]

Darcy's law: given g, κ, b find u

$$\begin{aligned} -\nabla \cdot (\kappa(x)\nabla u(x)) &= g(x) \quad \text{in } D \\ u(x) &= b(x) \quad \text{on } \partial D \end{aligned}$$

For general D, $\kappa(x)$ this cannot be solved analytically.

PN for PDEs

Darcy's law: given g, κ, b find u

$$\begin{aligned} -\nabla \cdot (\kappa(x) \nabla u(x)) &= g(x) \quad \text{in } D \\ u(x) &= b(x) \quad \text{on } \partial D \end{aligned}$$

For general D, $\kappa(x)$ this cannot be solved analytically.

The majority of PDE solvers produce an approximation like:

$$\hat{u}(x) = \sum_{i=1}^{N} w_i \phi_i(x)$$

We want to quantify the error from finite N probabilistically.

Inverse Problem: Given partial information of g, b, u find κ

$$
\begin{aligned}
-\nabla \cdot (\kappa(\boldsymbol{x}) \nabla u(\boldsymbol{x})) &= g(\boldsymbol{x}) \quad \text{in } D \\
u(\boldsymbol{x}) &= b(\boldsymbol{x}) \quad \text{on } \partial D
\end{aligned}
$$

PN for PDEs

Inverse Problem: Given partial information of g, b, u find κ

$$
\begin{aligned}
-\nabla \cdot (\kappa(\mathbf{x}) \nabla u(\mathbf{x})) &= g(\mathbf{x}) \quad \text{in } D \\
u(\mathbf{x}) &= b(\mathbf{x}) \quad \text{on } \partial D
\end{aligned}
$$

Bayesian Inverse Problem:

$$
\kappa \sim \Pi_\kappa \longrightarrow \boxed{\text{Data } u(\mathbf{x}_i) = y_i + \xi_i} \longrightarrow \kappa | \mathbf{y} \sim \Pi_\kappa^{\mathbf{y}}
$$

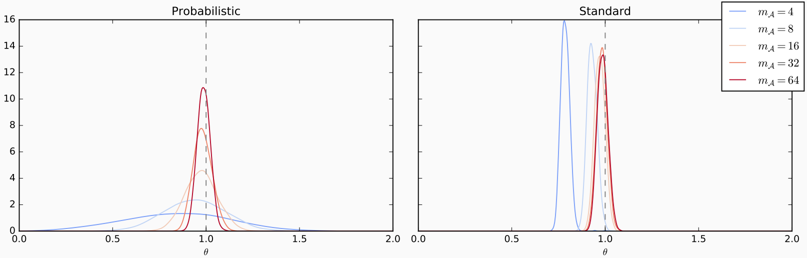
We want to account for an inaccurate forward solver in the inverse problem.

Why do this?

Using an inaccurate forward solver in an inverse problem can produce biased and overconfident posteriors.

Why do this?

Using an inaccurate forward solver in an inverse problem can produce biased and overconfident posteriors.



Comparison of inverse problem posteriors produced using the Probabilistic Meshless Method (PMM) vs. symmetric collocation.

Forward Problem

Abstract Formulation

$$\mathcal{A}u(\mathbf{x}) = g(\mathbf{x}) \qquad \text{in } D$$

Forward inference procedure:

$$u \sim \Pi_u \longrightarrow \boxed{\text{“Data” } \mathcal{A}u(\mathbf{x}_i) = g(\mathbf{x}_i)} \longrightarrow u|\mathbf{g} \sim \Pi_u^{\mathbf{g}}$$

Posterior for the forward problem

Use a Gaussian Process prior $u \sim \Pi_u = \mathcal{GP}(0, k)$. Assuming linearity, the posterior Π_u^g is available in closed-form[2].

[2] [Cockayne et al., 2016, Särkkä, 2011, Cialenco et al., 2012, Owhadi, 2014]

Posterior for the forward problem

Use a Gaussian Process prior $u \sim \Pi_u = \mathcal{GP}(0, k)$. Assuming linearity, the posterior $\Pi_u^{\boldsymbol{g}}$ is available in closed-form[2].

$$\Pi_u^{\boldsymbol{g}} \sim \mathcal{GP}(m_1, \Sigma_1)$$
$$m_1(\mathbf{x}) = \bar{\mathcal{A}}K(\mathbf{x}, X)\left[\mathcal{A}\bar{\mathcal{A}}K(X, X)\right]^{-1}\boldsymbol{g}$$
$$\Sigma_1(\mathbf{x}, \mathbf{x}') = k(\mathbf{x}, \mathbf{x}') - \bar{\mathcal{A}}K(\mathbf{x}, X)\left[\mathcal{A}\bar{\mathcal{A}}K(X, X)\right]^{-1}\mathcal{A}K(X, \mathbf{x}')$$

$\bar{\mathcal{A}}$ the adjoint of $\mathcal{A}$

Observation: The mean function is the same as in symmetric collocation!

[2] [Cockayne et al., 2016, Särkkä, 2011, Cialenco et al., 2012, Owhadi, 2014]

Theoretical Results

Theorem (Forward Contraction)

For a ball $B_\epsilon(u_0)$ of radius ϵ centered on the true solution u_0 of the PDE, we have

$$1 - \Pi_u^g[B_\epsilon(u_0)] = \mathcal{O}\left(\frac{h^{2\beta-2\rho-d}}{\epsilon}\right)$$

- *h the fill distance*
- *β the smoothness of the prior*[3]
- *$\rho < \beta - d/2$ the order of the PDE*
- *d the input dimension*

[3] Sobolev smoothness of a space norm-equivalent to the RKHS induced by the prior covariance.

Toy Example

$$\begin{aligned} -\nabla^2 u(x) &= g(x) \qquad & x \in (0,1) \\ u(x) &= 0 \qquad & x = 0, 1 \end{aligned}$$

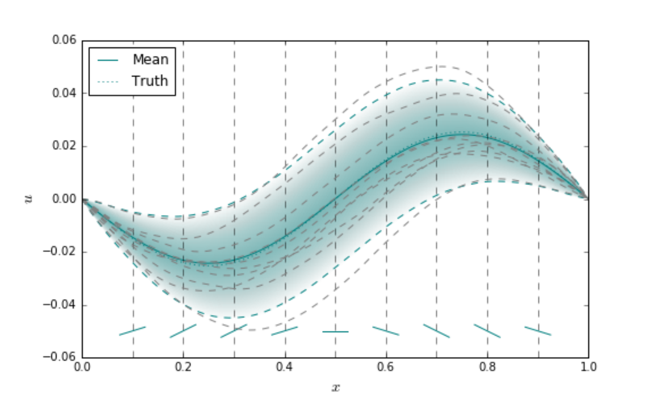
To associate with the notation from before...

$$\Pi_u \sim \mathcal{GP}(0, k(x,y))$$

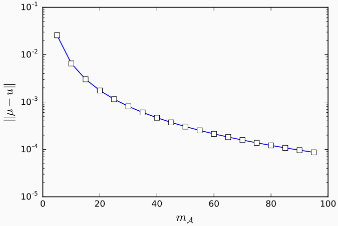
$$\mathcal{A} = -\frac{d^2}{dx^2} \quad \bar{\mathcal{A}} = -\frac{d^2}{dy^2}$$

Forward problem: posterior samples

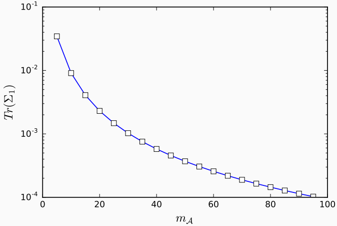
$$g(x) = \sin(2\pi x)$$



Forward problem: convergence



(a) Mean error from truth



(b) Trace of posterior covariance

Figure 2: Convergence

Inverse Problem

Recap

$$
\begin{aligned}
-\nabla \cdot (\kappa(\mathbf{x})\nabla u(\mathbf{x})) &= g(\mathbf{x}) \quad \text{in } D \\
u(\mathbf{x}) &= b(\mathbf{x}) \quad \text{on } \partial D
\end{aligned}
$$

Now we need to incorporate the forward posterior measure Π_u^g into the posterior measure for the inverse problem, κ

Incorporation of Forward Measure

Assuming the data in the inverse problem is:

$$y_i = u(\mathbf{x}_i) + \xi_i \quad i = 1, \ldots, n$$
$$\boldsymbol{\xi} \sim N(\mathbf{0}, \Gamma)$$

implies the standard likelihood:

$$p(\mathbf{y}|\kappa, u) \sim N(\mathbf{y}; \mathbf{u}, \Gamma)$$

But we don't know u

Incorporation of Forward Measure

Assuming the data in the inverse problem is:

$$y_i = u(\mathbf{x}_i) + \xi_i \quad i = 1, \dots, n$$
$$\boldsymbol{\xi} \sim N(\mathbf{0}, \Gamma)$$

implies the standard likelihood:

$$p(\mathbf{y} | \kappa, u) \sim N(\mathbf{y}; \mathbf{u}, \Gamma)$$

But we don't know u

Marginalise the forward posterior Π_u^g to obtain a "PN" likelihood:

$$p_{\mathrm{PN}}(\mathbf{y} | \kappa) \propto \int p(\mathbf{y} | \kappa, u) d\Pi_u^g$$
$$\sim N(\mathbf{y}; \mathbf{m}_1, \Gamma + \Sigma_1)$$

Inverse Contraction

Denote by Π^y_κ the posterior for κ from likelihood p, and by $\Pi^y_{\kappa,\mathrm{PN}}$ the posterior for κ from likelihood p_{PN}.

Theorem (Inverse Contraction)

Assume $\Pi^y_\kappa \to \delta(\kappa_0)$ *as* $n \to \infty$.

Then $\Pi^y_{\kappa,PN} \to \delta(\kappa_0)$ *provided*

$$h = o(n^{-1/(\beta-\rho-d/2)})$$

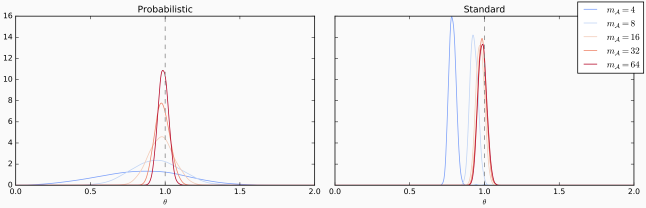
Back to the Toy Example

$$
\begin{aligned}
-\nabla \cdot (\kappa \nabla u(x)) &= \sin(2\pi x) \qquad & x \in (0,1) \\
u(x) &= 0 \qquad & x = 0, 1
\end{aligned}
$$

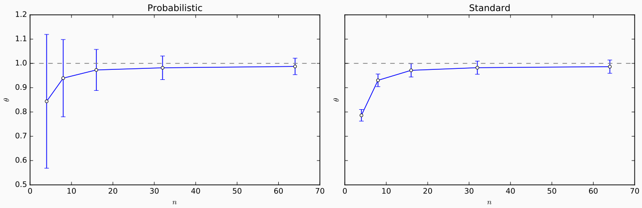
Infer $\kappa \in \mathbb{R}^+$; data generated for $\kappa = 1$ at $x = 0.25, 0.75$.

Corrupted with independent Gaussian noise $\xi \sim N(0, 0.01^2)$

Posteriors for κ



(a) Posterior Distributions for different numbers of design points.



(b) Convergence of posterior distributions with number of design points.

Nonlinear Example: Steady-State Allen–Cahn

Allen–Cahn

A prototypical nonlinear model.

$$
\begin{aligned}
-\delta \nabla^2 u(\boldsymbol{x}) + \delta^{-1}(u(\boldsymbol{x})^3 - u(\boldsymbol{x})) &= 0 \qquad && \boldsymbol{x} \in (0,1)^2 \\
u(\boldsymbol{x}) &= 1 && x_1 \in \{0,1\}\,;\, 0 < x_2 < 1 \\
u(\boldsymbol{x}) &= -1 && x_2 \in \{0,1\}\,;\, 0 < x_1 < 1
\end{aligned}
$$

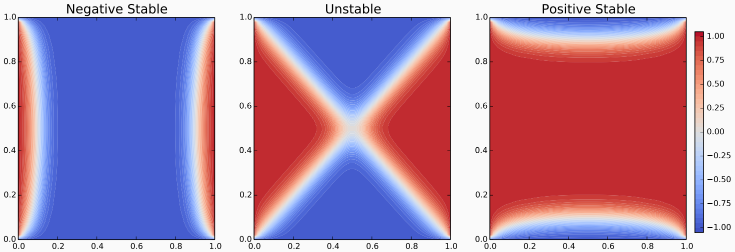
Goal: infer δ from 16 equally spaced observations of $u(\boldsymbol{x})$ in the interior of the domain.

Allen–Cahn

A prototypical nonlinear model.

$$
\begin{aligned}
-\delta\nabla^2 u(\boldsymbol{x}) + \delta^{-1}(u(\boldsymbol{x})^3 - u(\boldsymbol{x})) &= 0 & \boldsymbol{x} &\in (0,1)^2 \\
u(\boldsymbol{x}) &= 1 & x_1 &\in \{0,1\}\,;\, 0 < x_2 < 1 \\
u(\boldsymbol{x}) &= -1 & x_2 &\in \{0,1\}\,;\, 0 < x_1 < 1
\end{aligned}
$$

Goal: infer δ from 16 equally spaced observations of $u(\boldsymbol{x})$ in the interior of the domain.



Allen–Cahn: A Linearization Trick

Nonlinear PDE - must sample from the posterior.

Allen–Cahn: A Linearization Trick

Nonlinear PDE - must sample from the posterior.

$$-\delta\nabla^2 u(\mathbf{x}) + \delta^{-1}(u(\mathbf{x})^3 - u(\mathbf{x})) = 0 \tag{1}$$

split the operator...

$$-\delta\nabla^2 u(\mathbf{x}) - \delta^{-1} u(\mathbf{x}) = z \tag{2}$$

$$\delta^{-1} u(\mathbf{x})^3 = -z \tag{3}$$

(1) = (2) + (3)

Allen–Cahn: A Linearization Trick

Nonlinear PDE - must sample from the posterior.

$$-\delta\nabla^2 u(x) + \delta^{-1}(u(x)^3 - u(x)) = 0$$

...and invert

$$\begin{aligned} -\delta\nabla^2 u(x) - \delta^{-1}u(x) &= z \\ u(x) &= \sqrt[3]{-\delta z} \end{aligned}$$

Allen–Cahn: A Linearization Trick

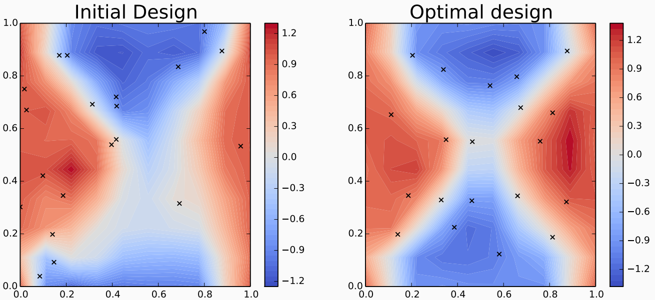
Nonlinear PDE - must sample from the posterior.

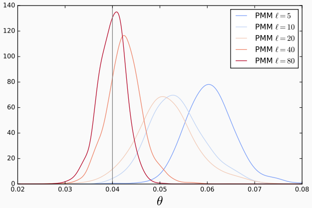
$\implies$ Solve the new system

$$
\begin{aligned}
\mathcal{A}_1 u(x) &:= -\delta \nabla^2 u(x) - \delta^{-1} u(x) &&= z \\
\mathcal{A}_2 u(x) &:= u(x) &&= \sqrt[3]{-\delta z}
\end{aligned}
$$

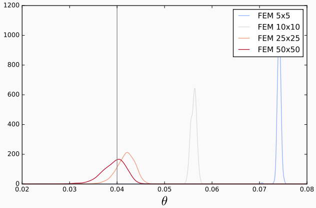
...only now we have a z

Experimental Design & Forward Solutions





(a) PMM



(b) FEA

Comparison of posteriors for δ with different solver resolutions, when using the PMM forward solver with PN likelihood, vs. FEA forward solver with Gaussian likelihood.

Conclusions

Conclusions

We have shown...

- How to build probability measures for the forward solution of PDEs.
- How to use this to make rhobust inferences in PDE inverse problems, even with inaccurate forward solvers.

All of this relies upon Gaussianity and linearity (or tricks to reduce to this).

Current Research

All of this relies upon Gaussianity and linearity (or tricks to reduce to this).

- How far can this be relaxed?
- When is a probabilistic numerical method “well posed”?
- How can we sample the posterior in this situation?

Current Research

All of this relies upon Gaussianity and linearity (or tricks to reduce to this).

- How far can this be relaxed?
- When is a probabilistic numerical method “well posed”?
- How can we sample the posterior in this situation?

…coming soon…

Thanks!

References

I. Cialenco, G. E. Fasshauer, and Q. Ye. Approximation of stochastic partial differential equations by a kernel-based collocation method. *Int. J. Comput. Math.*, 89(18):2543–2561, 2012.

J. Cockayne, C. J. Oates, T. Sullivan, and M. Girolami. Probabilistic Meshless Methods for Partial Differential Equations and Bayesian Inverse Problems. *arXiv preprint arXiv:1605.07811*, 2016.

P. Diaconis. Bayesian numerical analysis. *Statistical decision theory and related topics IV*, 1:163–175, 1988.

J. B. Kadane. Parallel and Sequential Computation: A Statistician's view. *Journal of Complexity*, 1:256–263, 1985.

A. O'Hagan. Some Bayesian numerical analysis. *Bayesian Statistics*, 4:345–363, 1992.

H. Owhadi. Bayesian numerical homogenization. *arXiv preprint arXiv:1406.6668*, 2014.

S. Särkkä. Linear operators and stochastic partial differential equations in Gaussian process regression. In *Artificial Neural Networks and Machine Learning – ICANN 2011: 21st International Conference on Artificial Neural Networks, Espoo, Finland, June 14-17, 2011, Proceedings, Part II*, pages 151–158. Springer, 2011.

J. Skilling. Bayesian solution of ordinary differential equations. *Maximum Entropy and Bayesian Methods*, 50:23–37, 1991.