

Variational Bayesian Approximation of Inverse Problems using Sparse Precision Matrices

Jan Povala¹ Ieva Kazlauskaitė¹ Eky Febrianto Fehmi Cirak
Mark Girolami

April 22, 2022

¹equal contribution

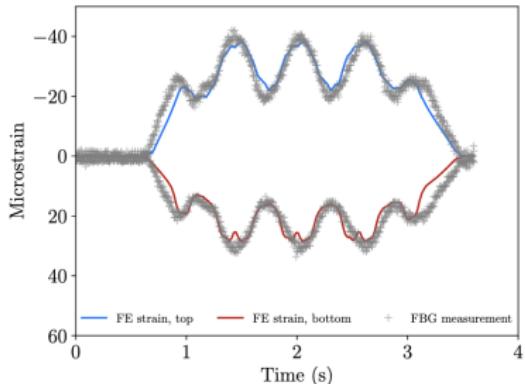
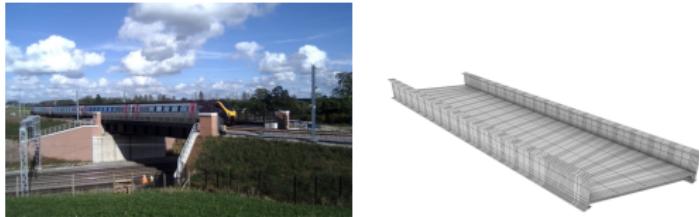
Partial Differential Equations

- ▶ Widely used in science and engineering.
- ▶ PDEs provide a mechanistic and interpretable way of specifying the relationship between different quantities.
- ▶ The partial derivatives encode how a quantity changes as we ‘move’ through the domain.
- ▶ Examples: fluid dynamics, elasticity of materials, heat systems, electrostatics, steady-state flow of groundwater through an aquifer.
- ▶ Thanks to advances in sensing, in-situ measurements have become widely available. How do we use it to improve our models?

Real-life Example

Febrianto, Butler, Girolami & Cirak (2021)

A railway bridge in Staffordshire which has been instrumented with fibre optic sensors (top left), its digital twin (top right), sample sensor measurements (bottom):



Using the Model and the Data

1. Using the model to answer ‘what if’ scenarios: How would the system behave under different conditions?
2. Using the measurements of the state of the system to infer the system’s properties.

The latter problem is termed the **inverse problem**.

The Rest of the Talk

1. Bayesian Inverse problem formulation.
2. Variational Bayes as an alternative to Markov Chain Monte Carlo methods.
3. Leveraging problem structure to specify the approximating family of distributions.
4. Results on an elliptic PDE.
5. Bimodal example.
6. Conclusion.

General Model (Stuart 2010)

- ▶ Given Banach spaces \mathcal{K}, \mathcal{Y} ², for $y \in \mathcal{Y}, \kappa \in \mathcal{K}$ we posit

$$y = \mathcal{G}(\kappa) + \eta, \quad (1)$$

where $\mathcal{G} : \mathcal{K} \rightarrow \mathcal{Y}, \eta \in \mathcal{Y}$ is additive observational noise.

- ▶ For a suitable Hilbert space \mathcal{U} , let $\mathcal{A} : \mathcal{K} \rightarrow \mathcal{U}$ be a possibly non-linear solution operator of a PDE. For a particular $\kappa \in \mathcal{K}$, the solution $u \in \mathcal{U}$ is

$$u = \mathcal{A}(\kappa). \quad (2)$$

- ▶ To obtain observation y , we define a projection operator $\mathcal{P} : \mathcal{U} \rightarrow \mathcal{Y}$. Consequently, (1) can be written out in full as

$$y = \mathcal{P}(\mathcal{A}(\kappa)) + \eta. \quad (3)$$

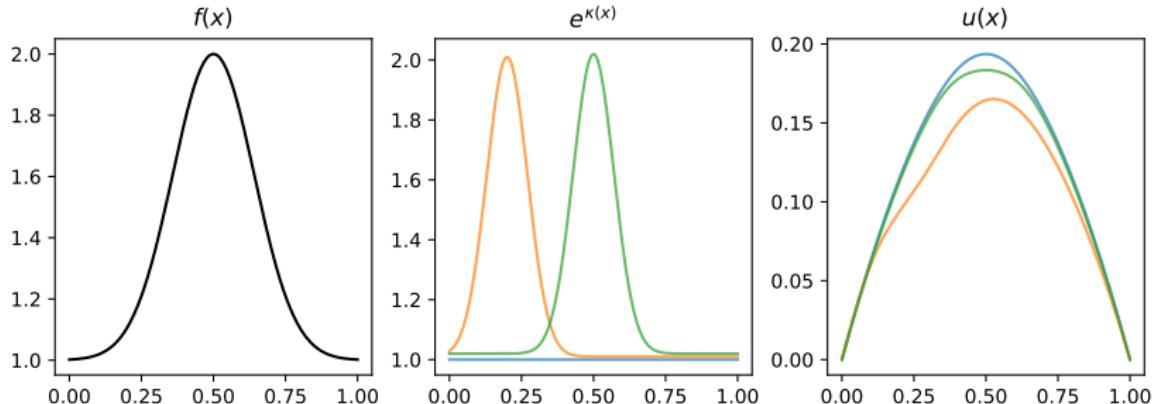
- ▶ Objective: want to find $\kappa \in \mathcal{K}$, the parameter to a model, given $y \in \mathcal{Y}$, a noisy observation of the solution of a PDE.

²Respective norms for Banach spaces \mathcal{K}, \mathcal{Y} are $\|\cdot\|_{\mathcal{K}}$ and $\|\cdot\|_{\mathcal{Y}}$.

Example – Poisson Equation

Given forcing f , log-diffusivity κ , and appropriate boundary conditions, we have:

$$-\nabla \cdot (\exp(\kappa(x)) \nabla u(x)) = f(x), \quad (4)$$



Inverse Problem: Finding κ

- ▶ General approach: finding κ such that the data misfit, $\|y - \mathcal{G}(\kappa)\|_{\mathcal{Y}}$, is minimised.
- ▶ This is an ill-posed problem: there may be **no solution**, it may **not** be **unique**, and it may depend **sensitively** on y .
- ▶ Assumptions about κ are implemented via some form of **regularisation**. Bayesian approach:
 1. Describe our prior knowledge about κ in terms of a prior probability measure μ_0 on the subspace of \mathcal{K} .
 2. Use Bayes' formula to calculate the posterior probability measure, μ^y , for κ given y .

Bayesian Inference I

- The relationship between the posterior and prior is expressed as

$$\frac{d\mu^y}{d\mu_0}(\kappa) = \frac{1}{Z(y)} \exp(-\Phi(\kappa; y)), \quad (5)$$

where $\frac{d\mu^y}{d\mu_0}$ is the Radon-Nikodym derivative of μ^y with respect to μ_0 , and Φ is the potential function which is determined by \mathcal{G} and η .
 $Z(y) = \int_{\mathcal{K}} \exp(-\Phi(\kappa; y)) d\mu_0(\kappa)$.

- From here on, we assume that $(\mathcal{Y}, \|\cdot\|_{\mathcal{Y}}) = (\mathbb{R}^{n_y}, \|\cdot\|)$, and we treat y and η as vectors, i.e. \mathbf{y} and $\boldsymbol{\eta}$. We specify the additive noise vector $\boldsymbol{\eta}$ as Gaussian white noise with standard deviation σ_y . We can write Φ conveniently as

$$\Phi(\kappa; \mathbf{y}) = \frac{1}{2} \|\mathcal{G}(\kappa) - \mathbf{y}\|_{\boldsymbol{\Gamma}^{-\frac{1}{2}}}^2, \quad (6)$$

where $\boldsymbol{\Gamma} = \sigma_y^2 \mathbf{I}$, and $\|\cdot\|_{\boldsymbol{\Gamma}}$ is the norm induced by the weighted inner product.

Bayesian Inference II

- ▶ Restrict the space of solutions \mathcal{K} to be a Hilbert space and place a Gaussian prior measure on κ with mean m and covariance operator \mathcal{C}_κ such that

$$\mu_0(\kappa) \sim \mathcal{N}(m, \mathcal{C}_\kappa). \quad (7)$$

- ▶ To proceed with inference, we have two options:
 1. Derive an inference scheme first, discretise afterwards
 2. Discretise first, derive an inference afterwards.

Inference in Infinite-dimensional Space

- ▶ Devise an inference scheme in the function space before discretising.
- ▶ Pre-conditioned Crank-Nicholson Markov Chain Monte Carlo (pCN) is a widely used algorithm (Cotter et al. 2013, Pinski et al. 2015).
- ▶ Recently, MCMC methods that leverage the geometry of the posterior have been proposed (Beskos et al. 2017).
- ▶ Early developments for variational Bayes in function space include: Minh (2017), Burt et al. (2021).

Discretise-first Approach

- ▶ Discretise the problem before deriving an inference scheme.
- ▶ Many widely used methods fall under this approach:
 - **Metropolis-Hastings** (Metropolis et al. 1953, Hastings 1970)
 - **Hamiltonian Monte Carlo** (HMC) (Duane et al. 1987),
 - **Laplace approximation**,
 - **Variational Bayes** (Jordan et al. 1998, 1999).
- ▶ This leads to the more familiar formulation of the Bayes' theorem:

$$p(\boldsymbol{\kappa} \mid \mathbf{y}) = \frac{p(\mathbf{y} \mid \boldsymbol{\kappa}) p(\boldsymbol{\kappa})}{p(\mathbf{y})} \propto p(\mathbf{y} \mid \boldsymbol{\kappa}) p(\boldsymbol{\kappa}), \quad (8)$$

where $p(\boldsymbol{\kappa} \mid \mathbf{y})$ is the posterior density, $p(\mathbf{y} \mid \boldsymbol{\kappa})$ is the likelihood of the observed data \mathbf{y} for a given discretised $\boldsymbol{\kappa}$.

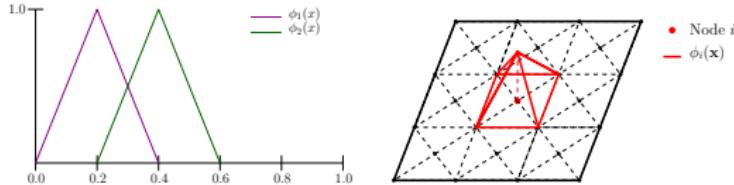
Likelihood – Back to the Poisson Problem I

- We consider a specific problem – the Poisson problem:

$$-\nabla \cdot (\exp(\kappa(\mathbf{x})) \nabla u(\mathbf{x})) = f(\mathbf{x}), \quad (9)$$

where $\mathbf{x} \in \Omega \subset \mathbb{R}^d$, with $d \in \{1, 2, 3\}$, $\kappa(\mathbf{x}) \in \mathbb{R}$ is the log-diffusion coefficient, $u(\mathbf{x}) \in \mathbb{R}$ is the unknown, and $f(\mathbf{x}) \in \mathbb{R}$ is a deterministic forcing term.

- Discretise the weak form of the problem with a standard **finite element approach** of non-overlapping elements $\{\omega_e\}_{e=1}^{n_e}$ and using Lagrange basis functions $\phi(\mathbf{x})$.



Likelihood – Back to the Poisson Problem II

- ▶ The solution u is then expressed using coefficients $\{u_i\}_{i=1}^{n_u}$ as

$$u_h(\mathbf{x}) = \sum_{i=1}^{n_u} \phi_i(\mathbf{x}) u_i. \quad (10)$$

- ▶ Proceeding further, we obtain a linear system:

$$\mathbf{A}(\boldsymbol{\kappa})\mathbf{u} = \mathbf{f}, \quad (11)$$

where $\mathbf{A}(\boldsymbol{\kappa}) \in \mathbb{R}^{n_u \times n_u}$ is the stiffness matrix, $\boldsymbol{\kappa} \in \mathbb{R}^{n_\kappa}$ is the vector of log-diffusion coefficients, $\mathbf{f} \in \mathbb{R}^{n_u}$ is the nodal source vector.

- ▶ The stiffness matrix of an element with label e is given by

$$A_{ij}^e(\kappa_e) = \int_{\omega_e} \exp(\kappa_e) \frac{\partial \phi_i(\mathbf{x})}{\partial \mathbf{x}} \cdot \frac{\partial \phi_j(\mathbf{x})}{\partial \mathbf{x}} d\mathbf{x}, \quad (12)$$

where the log-diffusion coefficient κ_e of the element is assumed to be *constant* within the element.

Likelihood – Back to the Poisson Problem III

- ▶ The source vector is discretised as:

$$f_i = \int_{\Omega} f(\boldsymbol{x}) \phi_i(\boldsymbol{x}) dx . \quad (13)$$

- ▶ Finally, the likelihood is given by

$$p(\boldsymbol{y} \mid \boldsymbol{\kappa}) = p(\boldsymbol{y} \mid \mathbf{u}(\boldsymbol{\kappa})) = \mathcal{N}(\mathbf{PA}(\boldsymbol{\kappa})^{-1}\mathbf{f}, \sigma_y^2 \mathbf{I}) , \quad (14)$$

where the matrix \mathbf{P} represents the discretisation of the observation operator \mathcal{P} .

Prior

- ▶ Placing a zero-mean Gaussian Process prior on κ gives

$$\kappa \sim \mathcal{GP}(0, k_{\text{SE}}(\cdot, \cdot)). \quad (15)$$

- ▶ Discretise κ by assuming $\kappa(\mathbf{x})$ is constant on each element of the mesh. Centroids of the finite elements are the training points of the GP, giving:

$$p(\boldsymbol{\kappa}) = \mathcal{N}(\mathbf{0}, \mathbf{K}_\psi(\mathbf{X}, \mathbf{X})). \quad (16)$$

- ▶ Parameters ψ are covariance function parameters such as length-scale and amplitude.

Stochastic Variational Bayes

We posit a family of distributions \mathcal{D}_q from which we choose the minimiser of the Kullback-Leibler divergence:

$$q^*(\kappa) = \arg \min_{q(\kappa) \in \mathcal{D}_q} \text{KL}(q(\kappa) \parallel p(\kappa \mid \mathbf{y})) \quad (17)$$

which is equivalent to maximising the evidence lower bound (ELBO):

$$q^*(\kappa) = \arg \max_{q(\kappa) \in \mathcal{D}_q} \mathbb{E}_q [\log p(\mathbf{y} \mid \kappa)] - \text{KL}(q(\kappa) \parallel p(\kappa)). \quad (18)$$

Generally, $\mathbb{E}_q [\log p(\mathbf{y} \mid \kappa)]$ is not available in closed form. Instead, we use a Monte Carlo approximation with N_{SVI} samples from $q(\kappa)$ as follows

$$\mathbb{E}_q [\log p(\mathbf{y} \mid \kappa)] \approx N_{\text{SVI}}^{-1} \sum_{i=1}^{N_{\text{SVI}}} \log p(\mathbf{y} \mid \kappa^{(i)}), \quad (19)$$

where $\kappa^{(i)}$ is the i -th sample from $q(\kappa)$.

Choice of Variational Family

- ▶ The choice of the approximating family of distributions determines how much structure of the true posterior distribution is captured by the variational approximation.
- ▶ The family of Gaussian distributions is a convenient choice (the KL term of the ELBO is available in closed form for GP priors).
- ▶ Other options:
 - Mixture of variational densities to cater for multi-modal posteriors (Bishop et al. 1998).
 - embedding parameters of a mean-field approximation in a hierarchical model to induce variational dependencies between latent variables (Tran et al. 2015, Ranganath et al. 2016).

Multivariate Gaussian Parametrisations I

Different parametrisations of $q(\boldsymbol{\kappa})$ for the family of multivariate Gaussian distributions:

1. Diagonal covariance matrix, known as mean-field VB [MFVB]

$$q(\boldsymbol{\kappa}) \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{D})$$

2. Fully-specified Cholesky factor. (full-covariance VB [FCVB])

$$q(\boldsymbol{\kappa}) \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{L}\mathbf{L}^\top)$$

3. Cholesky factor of the sparse precision matrix [PMVB]

$$q(\boldsymbol{\kappa}) \sim \mathcal{N}(\boldsymbol{\mu}, (\mathbf{L}_Q \mathbf{L}_Q^\top)^{-1})$$

Precision Matrices and Conditional Independence

- ▶ Specify the multivariate Gaussian through the precision matrix:
$$Q = \Sigma^{-1}.$$
- ▶ The elements of the precision matrix reflect **conditional independence**:

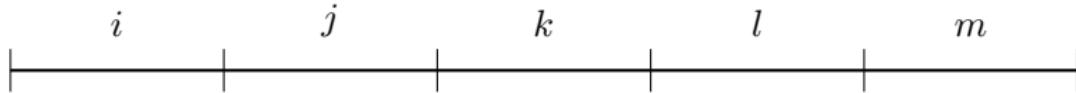
$$p(\kappa_i, \kappa_j \mid \kappa_{-\{i,j\}}) = p(\kappa_i \mid \kappa_{-\{i,j\}})p(\kappa_j \mid \kappa_{-\{i,j\}}) \Leftrightarrow Q_{ij} = 0, \quad (20)$$

whereas for **independence** (of Gaussian random variables!), we have:

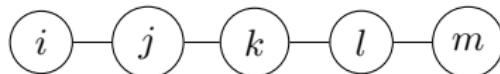
$$p(\kappa_i, \kappa_j) = p(\kappa_i)p(\kappa_j) \Leftrightarrow \Sigma_{ij} = 0. \quad (21)$$

- ▶ For more details, see Rue & Held (2005).

Leveraging Sparsity 1D Example

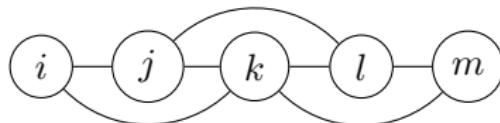


(a) Labelling of the five elements.



	i	j	k	l	m
i	\times	\times			
j	\times	\times	\times		
k		\times	\times	\times	
l			\times	\times	\times
m				\times	\times

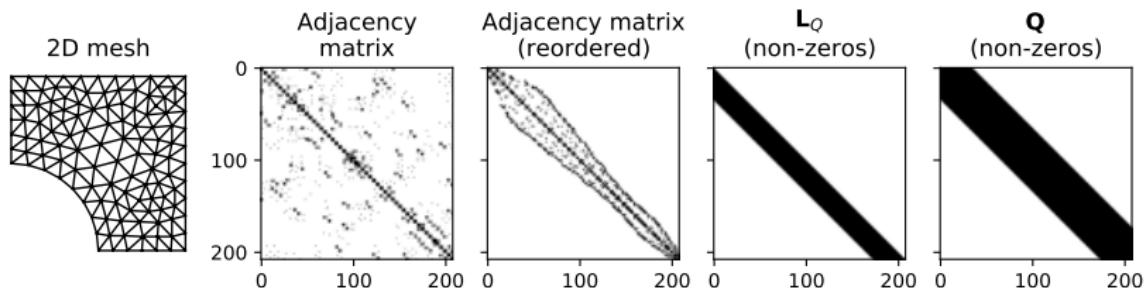
(b) 1-neighbourhood structure



	i	j	k	l	m
i	\times	\times	\times		
j	\times	\times	\times	\times	
k	\times	\times	\times	\times	\times
l		\times	\times	\times	\times
m			\times	\times	\times

(c) 2-neighbourhood structure

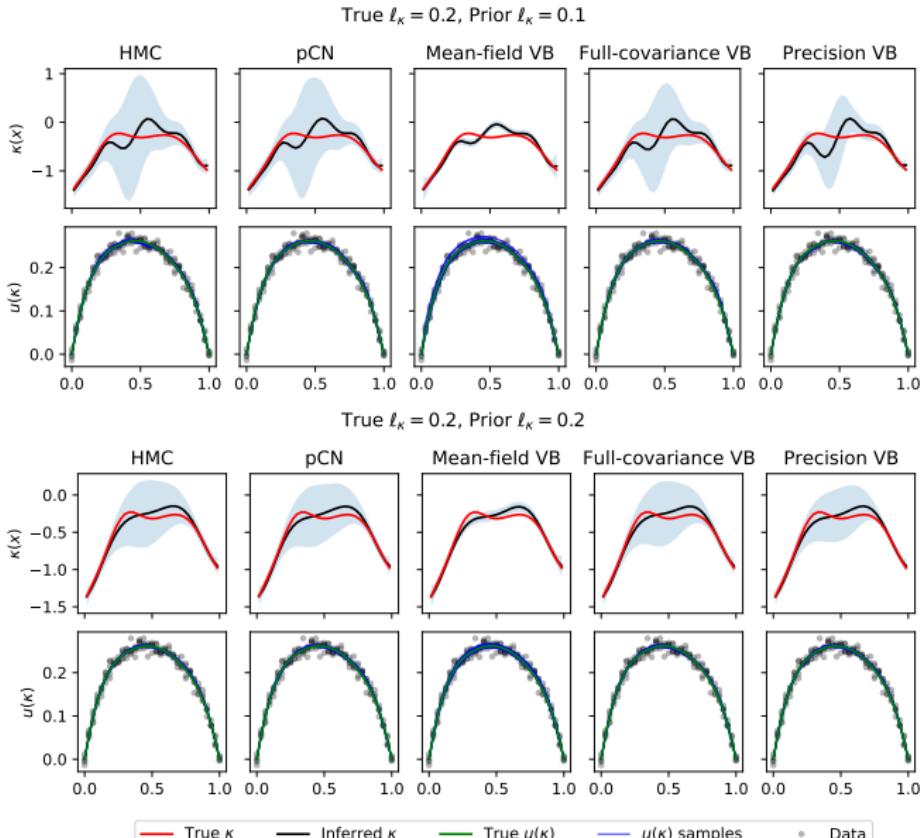
Leveraging Sparsity 2D Example



Experiments – 1D Poisson Problem

- ▶ Objective: infer the posterior distribution of κ given measurements of y .
- ▶ Sensors are placed on the discretisation nodes and the sensor noise is $\sigma_y = 0.01$. Five readings are taken.
- ▶ Constant forcing, $f(x) = 1$ is assumed and we impose Dirichlet boundary conditions as $u(0) = u(1) = 0$.
- ▶ Data generated by assuming κ is a sample from a unit-variance zero-mean GP with lengthscale $\ell_\kappa = 0.2$.

Results – Uncertainty Estimate



Results – Uncertainty Propagation

We assess the propagation of uncertainty using log of total boundary flux through the left boundary:

$$r(\kappa) = \log \int_{\Gamma_b} e^{\kappa(s)} \nabla u(s) \cdot \mathbf{n} \, ds, \quad (22)$$

where \mathbf{n} is a unit vector normal to the boundary.

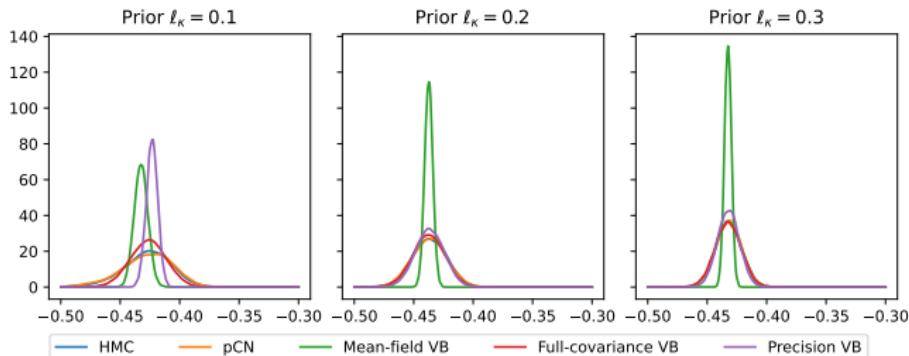
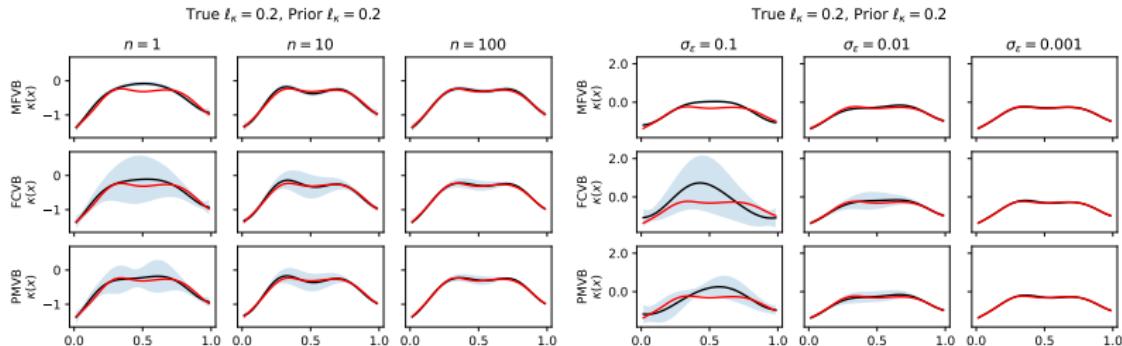


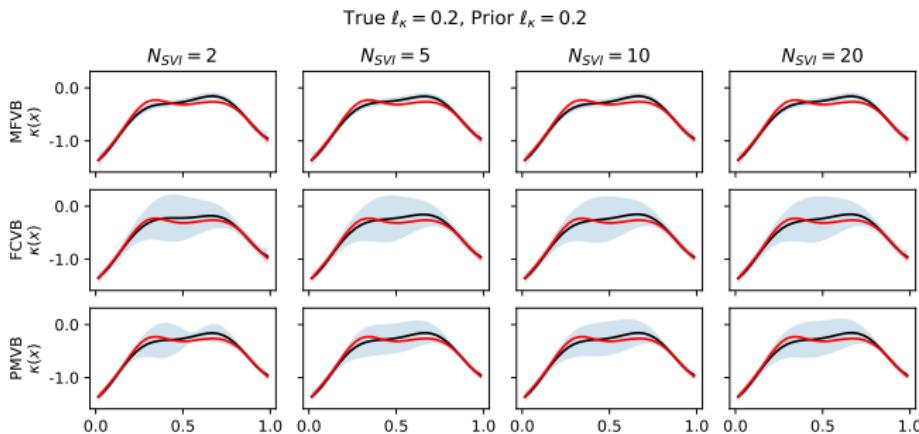
Figure: Log of the boundary flux at the left boundary node ($x = 0$) for the 1D Poisson example. For PMVB, the precision matrix bandwidth of 10 is used.

Results – Data Quality



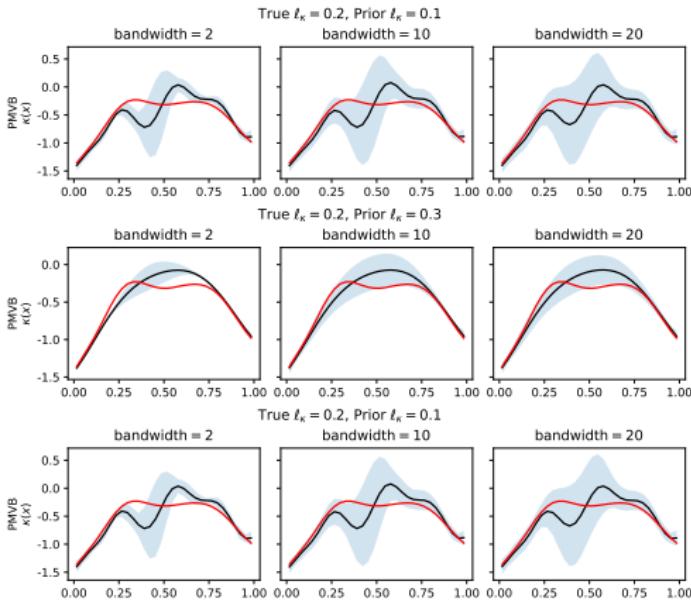
- ▶ Mean estimates for all methods get closer to the true κ with improved information.
- ▶ FCVB and PMVB uncertainty estimates get narrower with increasing number of observations and with decreasing observational noise.

Results – Number of Samples for VB



- ▶ Figure shows the posterior estimates for different number of Monte Carlo samples in the estimation of ELBO,
- ▶ On a qualitative level, a small number of samples is sufficient to obtain a good estimate.

Results – Increasing PMVB Bandwidth



- ▶ Increasing the order of neighbourhood improves the estimation of uncertainty (at the cost of more parameters).

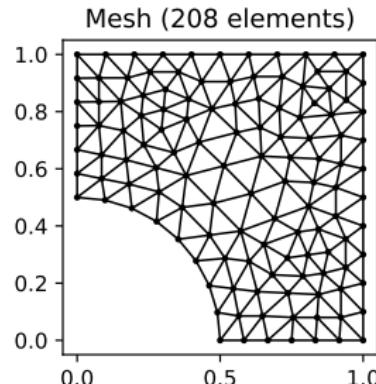
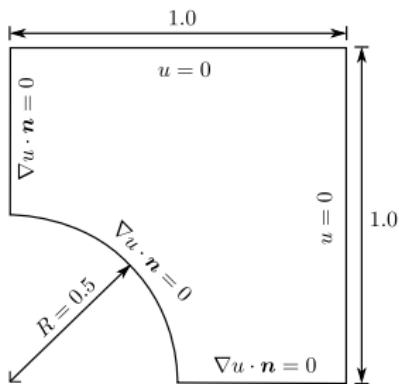
Results – Computation Time

true ℓ_κ	prior ℓ_κ	Time (hours)			
		HMC	MFVB	FCVB	PMVB
0.1	0.1	15.2 (871–3244)	1.1	3.6	2.1
	0.2	11.1 (1043–4006)	0.7	2.7	2.1
	0.3	7.2 (1130–5408)	0.6	2.3	2.0
0.2	0.1	15.2 (1600–4700)	0.6	2.2	1.8
	0.2	10.4 (1067–3468)	0.6	2.3	2.0
	0.3	7.0 (1487–3969)	0.5	1.7	1.8

Table: Run-times in hours for the Poisson 1D problem. For VB methods, $N_{\text{SVI}} = 3$. The column for HMC includes the range of effective sample sizes (ESS) across different components of κ .

Experiments – 2D Poisson Problem

- ▶ Same setup as before, but now with the sensor noise $\sigma_y = 0.001$, and different boundary conditions.
- ▶ Dirichlet boundary conditions $u(x, y) = 0$ when $x = 1$ or $y = 1$. Neumann boundary conditions on the rest of the boundary



2D Example – Results I

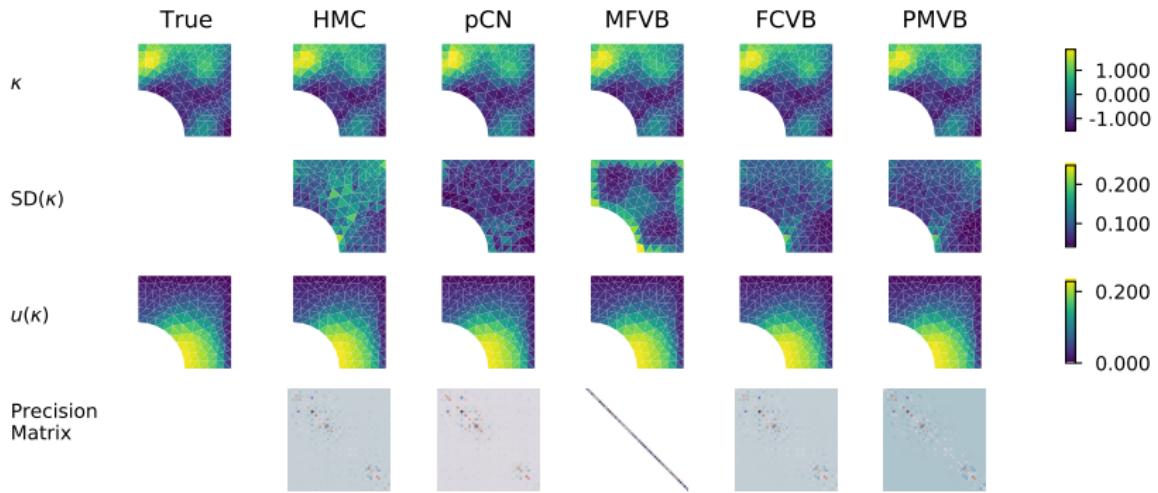


Figure: Posterior mean and standard deviation for κ and the corresponding u for 2D Poisson example with prior length-scale $\ell_\kappa = 0.1$. The bottom row shows the structure of the precision matrix for each inference scheme.

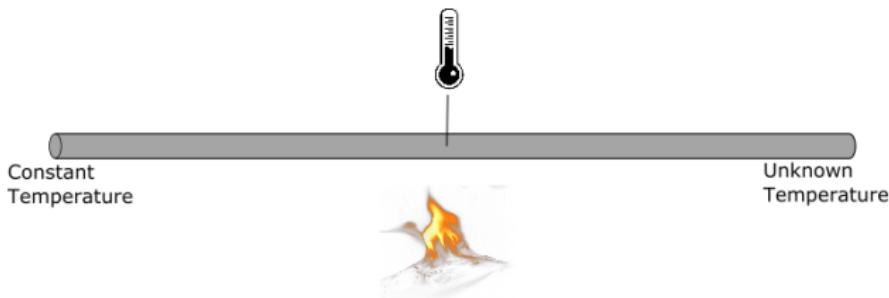
2D Example –Computation Cost

true ℓ_κ	prior ℓ_κ	Time (hours)				
		HMC	MFVB	FCVB	PMVB	
0.1	0.1	240.6 (930–11200)	6.4	29.6	28.1	
	0.2	295.5 (1537–11067)	6.6	32.6	28.9	
	0.3	242.0 (1057–6068)	7.3	27.3	30.6	
0.2	0.1	242.7 (1102–18235)	6.2	34.3	27.2	
	0.2	264.3 (1304–9848)	7.4	33.7	34.0	
	0.3	221.9 (1192–6356)	7.8	31.3	34.0	

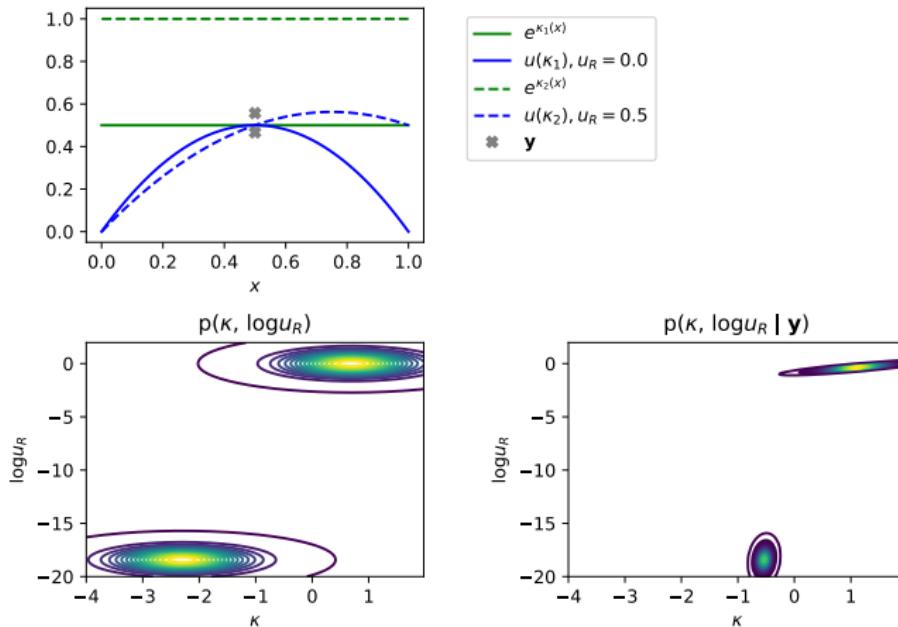
Table: Run-times for different inference schemes in seconds. The number of Monte Carlo samples is $N_{\text{SVI}} = 5$ for all MFVB, FCVB, and PMVB. The column for HMC includes the range of effective sample sizes (ESS) across different components of κ .

Heat Example

- ▶ Metal rod with *unknown* conductivity properties (constant throughout).
- ▶ A uniform heat source throughout the rod.
- ▶ Temperature on the RHS is fixed and *unknown*, and on the LHS is fixed and known (BC).
- ▶ We obtain two readings of temperature in the centre of the rod.
- ▶ What can we say about heat conductivity, and the temperature on the RHS which is fixed?



VB Can Handle Multi-modal Posteriors



Implementation

- ▶ FEM C++ code written by CSMLab, led by Prof Fehmi Cirak.
- ▶ Our Tensorflow (Python) code performs the inference and interfaces with the FEM module.
- ▶ Available on Github.
- ▶ For more details, see the paper: Povala et al. (2022)

Conclusions

- ▶ the mean of the variational posterior provides an accurate point estimate irrespective of the choice of the parametrisation of the covariance structure,
- ▶ VB with a full-covariance or precision matrix structure adequately estimates posterior uncertainty compared to HMC and pCN,
- ▶ sparse precision matrix parameterisation leverages the structure of the problem to balance computational complexity and the ability to capture dependencies in the posterior distribution,
- ▶ VB provides a good estimate for mean and variance in a time that is at least an order of magnitude faster than HMC or pCN,
- ▶ the VB estimates may be used effectively in downstream tasks to estimate various quantities of interest.

Outlook:

- ▶ Leverage sparse linear algebra routines and different optimisation schemes.
- ▶ Consider stochastic forcing.

Bibliography I

- Beskos, A., Girolami, M., Lan, S., Farrell, P. E. & Stuart, A. M. (2017), 'Geometric MCMC for infinite-dimensional inverse problems', *Journal of Computational Physics* **335**, 327–351.
- Bishop, C., Lawrence, N., Jaakkola, T. & Jordan, M. (1998), Approximating posterior distributions in belief networks using mixtures, *in* M. Jordan, M. Kearns & S. Solla, eds, 'Advances in Neural Information Processing Systems', Vol. 10, MIT Press.
- Burt, D. R., Ober, S. W., Garriga-Alonso, A. & van der Wilk, M. (2021), Understanding variational inference in function-space, *in* 'Third Symposium on Advances in Approximate Bayesian Inference'.
- Cotter, S. L., Roberts, G. O., Stuart, A. M. & White, D. (2013), 'MCMC methods for functions: Modifying old algorithms to make them faster', *Statistical Science* **28**(3), 424–446.
- Duane, S., Kennedy, A. D., Pendleton, B. J. & Roweth, D. (1987), 'Hybrid Monte Carlo', *Physics Letters B* **195**(2), 216–222.
- Febrianto, E., Butler, L., Girolami, M. & Cirak, F. (2021), 'Digital twinning of self-sensing structures using the statistical finite element method'.

Bibliography II

- Hastings, W. K. (1970), 'Monte Carlo sampling methods using Markov chains and their applications', *Biometrika* **57**(1), 97–109.
- Jordan, M. I., Ghahramani, Z., Jaakkola, T. S. & Saul, L. K. (1999), 'An introduction to variational methods for graphical models', *Machine Learning* **37**(2), 183–233.
- Jordan, R., Kinderlehrer, D. & Otto, F. (1998), 'The variational formulation of the Fokker-Planck equation', *Siam Journal on Mathematical Analysis* **29**(1), 1–17.
- Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. & Teller, E. (1953), 'Equation of state calculations by fast computing machines', *The Journal of Chemical Physics* **21**(6), 1087–1092.
- Minh, H. Q. (2017), 'Infinite-dimensional log-determinant divergences between positive definite trace class operators', *Linear Algebra and its Applications* **528**, 331–383.
- Pinski, F. J., Simpson, G., Stuart, A. M. & Weber, H. (2015), 'Algorithms for Kullback–Leibler approximation of probability measures in infinite dimensions', *SIAM Journal on Scientific Computing* **37**(6), A2733–A2757.

Bibliography III

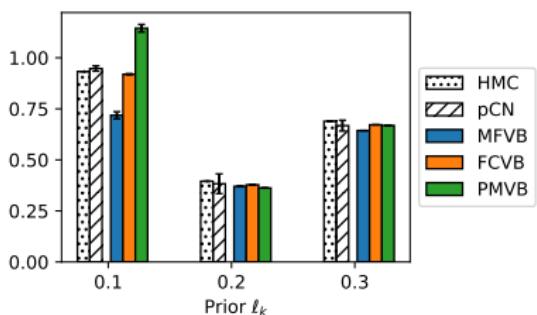
- Povala, J., Kazlauskaitė, I., Febrianto, E., Cirak, F. & Girolami, M. (2022), 'Variational Bayesian approximation of inverse problems using sparse precision matrices', *Computer Methods in Applied Mechanics and Engineering* **393**, 114712.
- Ranganath, R., Tran, D. & Blei, D. (2016), Hierarchical variational models, in M. F. Balcan & K. Q. Weinberger, eds, 'Proceedings of the 33rd International Conference on Machine Learning', Vol. 48 of *Proceedings of Machine Learning Research*, PMLR, New York, New York, USA, pp. 324–333.
- Rue, H. & Held, L. (2005), *Gaussian Markov Random Fields: Theory and Applications*, number 104 in 'Monographs on Statistics and Applied Probability', Chapman & Hall/CRC, Boca Raton.
- Stuart, A. M. (2010), 'Inverse problems: A Bayesian perspective', *Acta Numerica* **19**, 451–559.
- Tran, D., Blei, D. & Airoldi, E. M. (2015), Copula variational inference, in C. Cortes, N. Lawrence, D. Lee, M. Sugiyama & R. Garnett, eds, 'Advances in Neural Information Processing Systems', Vol. 28, Curran Associates, Inc.

Results – Error

$$\left\| \frac{1}{S} \sum_{s=1}^S \boldsymbol{\kappa}^{(s)} - \boldsymbol{\kappa}_{\text{true}} \right\|_2$$

$$\frac{1}{S} \sum_{s=1}^S \left\| \mathbf{u}(\boldsymbol{\kappa}^{(s)}) - \mathbf{u}(\boldsymbol{\kappa}_{\text{true}}) \right\|_2$$

Mean κ error norm
true $\ell_\kappa = 0.2$



Expected $u(\kappa)$ error norm
true $\ell_\kappa = 0.2$

