# Projected Stein variational Newton:
## A fast and scalable Bayesian inference method in high dimensions

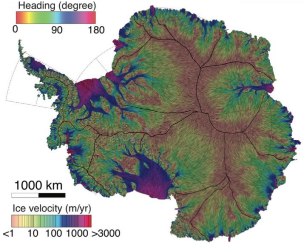

Peng Chen

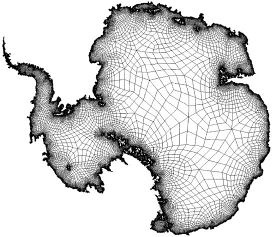
Keyi Wu, Joshua Chen, Thomas OLeary-Roseberry, Omar Ghattas

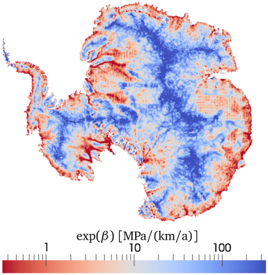
Oden Institute for Computational Engineering and Sciences
The University of Texas at Austin


RICAM Workshop on Optimization and Inversion under Uncertainty

# Example: inversion in Antarctica ice sheet flow






- uncertain parameter: basal sliding field in boundary condition
- forward model: viscous, shear-thinning, incompressible fluid

$$-\nabla \cdot (\eta(\boldsymbol{u})(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T) - \mathbf{I}p) = \rho \boldsymbol{g}$$

$$\nabla \cdot \boldsymbol{u} = 0$$

- data: (InSAR) satellite observation of surface ice flow velocity

T. Isaac, N. Petra, G. Stadler, O. Ghattas, JCP, 2015

# Outline

1. Bayesian inversion
2. Stein variational methods
3. Projected Stein variational methods
4. Stein variational reduced basis methods

# Outline

# Uncertainty parametrization

## Example I: Karhunen–Loève expansion

**Karhunen–Loève** expansion for $\beta$ with mean $\bar{\beta}$ and covariance $\mathcal{C}$

$$\beta(x, \theta) = \bar{\beta}(x) + \sum_{j \geq 1} \sqrt{\lambda_j} \psi_j(x) \; \theta_j,$$

$(\lambda_j, \psi_j)_{j \geq 1}$: eigenpairs of a covariance $\mathcal{C}$, $\theta = (\theta_j)_{j \geq 1}$, uncorrelated, given by

$$\theta_j = \frac{1}{\sqrt{\lambda_j}} \int_D (\kappa - \bar{\kappa}) \psi_j(x) dx.$$

## Example II: dictionary basis representation

We can approximate the random field $\beta$ by

$$\beta(x, \theta) = \sum_{j \geq 1} \psi_j(x) \; \theta_j,$$

$(\psi_j)_{j \geq 1}$ dictionary basis, e.g., wavelet or finite element basis.

# Bayesian inversion

We consider an abstract form of the parameter to data model

$$y = \mathcal{O}(\theta) + \xi$$

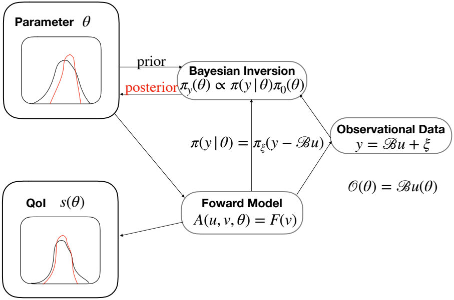
- uncertain parameter: $\theta \in \Theta \subset \mathbb{R}^d$
- parameter-to-observable map $\mathcal{O}$
- observation data: $y \in \mathbb{R}^n$
- noise $\xi$, e.g., $\xi \sim \mathcal{N}(0, \Gamma)$

Bayes' rule:

$$\underbrace{\pi_y(\theta)}_{\text{posterior}} = \frac{1}{\pi(y)} \underbrace{\pi(y|\theta)}_{\text{likelihood}} \underbrace{\pi_0(\theta)}_{\text{prior}},$$

with the model evidence

$$\pi(y) = \int_\Theta \pi(y|\theta)\pi_0(\theta)d\theta.$$



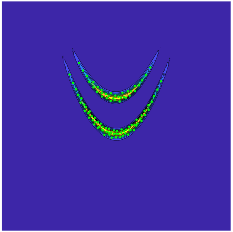

The central tasks: sample from posterior and compute statistics, e.g.,

$$\mathbb{E}_{\pi_y}[s] = \int_\Theta s(\theta)\pi_y(\theta)d\theta.$$

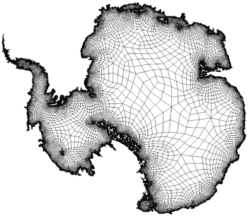
# Computational challenges

## Computational challenges for Bayesian inversion:

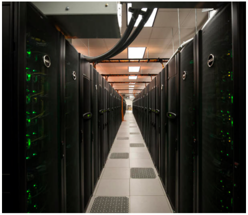
- the posterior has **complex** geometry:
  non-Gaussian, multimodal, concentrating in a local region
- the parameter lives in **high-dimensional** spaces
  curse of dimensionality – complexity grows exponentially
- the map $\mathcal{O}$ is **expensive** to evaluate:
  involving solve of large-scale partial differential equations



complex geometry



high dimensionality



large-scale computation

# Computational methods

- **Towards better design of MCMC to reduce # samples**
  1. Langevin and Hamiltonian MCMC (local geometry using gradient, Hessian, etc.) [Stuart et al., 2004, Girolami and Calderhead, 2011, Martin et al., 2012, Bui-Thanh and Girolami, 2014, Lan et al., 2016, Beskos et al., 2017]...
  2. dimension reduction MCMC (intrinsic low-dimensionality) [Cui et. al., 2014, 2016, Constantine et. al., 2016]...
  3. randomized/optimized MCMC (optimization for sampling) [Oliver, 2017, Wang et al., 2018, Wang et al., 2019]...
- **Direct posterior construction and statistical computation**
  1. Laplace approximation (Gaussian posterior approximation) [Bui-Thanh et al., 2013, Chen et al., 2017, Schillings et al., 2019]...
  2. deterministic quadrature (sparse Smolyak, high-order quasi-MC) [Schillings and Schwab, 2013, Gantner and Schwab, 2016, Chen and Schwab, 2016, Chen et al., 2017]...
  3. transport maps (polynomials, radial basis functions, deep neural networks) [El Moselhy and Marzouk, 2012, Spantini et al., 2018, Rezende and Mohamed, 2015, Liu and Wang, 2016, Detommaso et al., 2018, Chen et al., 2019]...

# Computational methods

- **Towards better design of MCMC to reduce # samples**
  1. Langevin and Hamiltonian MCMC (local geometry using gradient, Hessian, etc.) [Stuart et al., 2004, Girolami and Calderhead, 2011, Martin et al., 2012, Bui-Thanh and Girolami, 2014, Lan et al., 2016, Beskos et al., 2017]...
  2. dimension reduction MCMC (intrinsic low-dimensionality) [Cui et. al., 2014, 2016, Constantine et. al., 2016]...
  3. randomized/optimized MCMC (optimization for sampling) [Oliver, 2017, Wang et al., 2018, Wang et al., 2019]...
- **Direct posterior construction and statistical computation**
  1. Laplace approximation (Gaussian posterior approximation) [Bui-Thanh et al., 2013, Chen et al., 2017, Schillings et al., 2019]...
  2. deterministic quadrature (sparse Smolyak, high-order quasi-MC) [Schillings and Schwab, 2013, Gantner and Schwab, 2016, Chen and Schwab, 2016, Chen et al., 2017]...
  3. transport maps (polynomials, radial basis functions, deep neural networks) [El Moselhy and Marzouk, 2012, Spantini et al., 2018, Rezende and Mohamed, 2015, Liu and Wang, 2016, Detommaso et al., 2018, Chen et al., 2019]...

# Computational methods

- **Towards better design of MCMC to reduce # samples**
  1. Langevin and Hamiltonian MCMC (local geometry using gradient, Hessian, etc.) [Stuart et al., 2004, Girolami and Calderhead, 2011, Martin et al., 2012, Bui-Thanh and Girolami, 2014, Lan et al., 2016, Beskos et al., 2017]...
  2. dimension reduction MCMC (intrinsic low-dimensionality) [Cui et. al., 2014, 2016, Constantine et. al., 2016]...
  3. randomized/optimized MCMC (optimization for sampling) [Oliver, 2017, Wang et al., 2018, Wang et al., 2019]...
- **Direct posterior construction and statistical computation**
  1. Laplace approximation (Gaussian posterior approximation) [Bui-Thanh et al., 2013, Chen et al., 2017, Schillings et al., 2019]...
  2. deterministic quadrature (sparse Smolyak, high-order quasi-MC) [Schillings and Schwab, 2013, Gantner and Schwab, 2016, Chen and Schwab, 2016, Chen et al., 2017]...
  3. transport maps (polynomials, radial basis functions, deep neural networks) [El Moselhy and Marzouk, 2012, Spantini et al., 2018, Rezende and Mohamed, 2015, Liu and Wang, 2016, Detommaso et al., 2018, Chen et al., 2019]...

# Computational methods

- **Towards better design of MCMC to reduce # samples**
  1. Langevin and Hamiltonian MCMC (local geometry using gradient, Hessian, etc.) [Stuart et al., 2004, Girolami and Calderhead, 2011, Martin et al., 2012, Bui-Thanh and Girolami, 2014, Lan et al., 2016, Beskos et al., 2017]...
  2. dimension reduction MCMC (intrinsic low-dimensionality) [Cui et. al., 2014, 2016, Constantine et. al., 2016]...
  3. randomized/optimized MCMC (optimization for sampling) [Oliver, 2017, Wang et al., 2018, Wang et al., 2019]...
- **Direct posterior construction and statistical computation**
  1. Laplace approximation (Gaussian posterior approximation) [Bui-Thanh et al., 2013, Chen et al., 2017, Schillings et al., 2019]...
  2. deterministic quadrature (sparse Smolyak, high-order quasi-MC) [Schillings and Schwab, 2013, Gantner and Schwab, 2016, Chen and Schwab, 2016, Chen et al., 2017]...
  3. transport maps (polynomials, radial basis functions, deep neural networks) [El Moselhy and Marzouk, 2012, Spantini et al., 2018, Rezende and Mohamed, 2015, Liu and Wang, 2016, Detommaso et al., 2018, Chen et al., 2019]...

# Computational methods

- **Towards better design of MCMC to reduce # samples**
  1. Langevin and Hamiltonian MCMC (local geometry using gradient, Hessian, etc.) [Stuart et al., 2004, Girolami and Calderhead, 2011, Martin et al., 2012, Bui-Thanh and Girolami, 2014, Lan et al., 2016, Beskos et al., 2017]...
  2. dimension reduction MCMC (intrinsic low-dimensionality) [Cui et. al., 2014, 2016, Constantine et. al., 2016]...
  3. randomized/optimized MCMC (optimization for sampling) [Oliver, 2017, Wang et al., 2018, Wang et al., 2019]...
- **Direct posterior construction and statistical computation**
  1. Laplace approximation (Gaussian posterior approximation) [Bui-Thanh et al., 2013, Chen et al., 2017, Schillings et al., 2019]...
  2. deterministic quadrature (sparse Smolyak, high-order quasi-MC) [Schillings and Schwab, 2013, Gantner and Schwab, 2016, Chen and Schwab, 2016, Chen et al., 2017]...
  3. transport maps (polynomials, radial basis functions, deep neural networks) [El Moselhy and Marzouk, 2012, Spantini et al., 2018, Rezende and Mohamed, 2015, Liu and Wang, 2016, Detommaso et al., 2018, Chen et al., 2019]...

# Computational methods

- **Towards better design of MCMC to reduce # samples**
  1. Langevin and Hamiltonian MCMC (local geometry using gradient, Hessian, etc.) [Stuart et al., 2004, Girolami and Calderhead, 2011, Martin et al., 2012, Bui-Thanh and Girolami, 2014, Lan et al., 2016, Beskos et al., 2017]...
  2. dimension reduction MCMC (intrinsic low-dimensionality) [Cui et. al., 2014, 2016, Constantine et. al., 2016]...
  3. randomized/optimized MCMC (optimization for sampling) [Oliver, 2017, Wang et al., 2018, Wang et al., 2019]...
- **Direct posterior construction and statistical computation**
  1. Laplace approximation (Gaussian posterior approximation) [Bui-Thanh et al., 2013, Chen et al., 2017, Schillings et al., 2019]...
  2. deterministic quadrature (sparse Smolyak, high-order quasi-MC) [Schillings and Schwab, 2013, Gantner and Schwab, 2016, Chen and Schwab, 2016, Chen et al., 2017]...
  3. transport maps (polynomials, radial basis functions, deep neural networks) [El Moselhy and Marzouk, 2012, Spantini et al., 2018, Rezende and Mohamed, 2015, Liu and Wang, 2016, Detommaso et al., 2018, Chen et al., 2019]...

# Computational methods

- **Towards better design of MCMC to reduce # samples**
  1. Langevin and Hamiltonian MCMC (local geometry using gradient, Hessian, etc.) [Stuart et al., 2004, Girolami and Calderhead, 2011, Martin et al., 2012, Bui-Thanh and Girolami, 2014, Lan et al., 2016, Beskos et al., 2017]...
  2. dimension reduction MCMC (intrinsic low-dimensionality) [Cui et. al., 2014, 2016, Constantine et. al., 2016]...
  3. randomized/optimized MCMC (optimization for sampling) [Oliver, 2017, Wang et al., 2018, Wang et al., 2019]...
- **Direct posterior construction and statistical computation**
  1. Laplace approximation (Gaussian posterior approximation) [Bui-Thanh et al., 2013, Chen et al., 2017, Schillings et al., 2019]...
  2. deterministic quadrature (sparse Smolyak, high-order quasi-MC) [Schillings and Schwab, 2013, Gantner and Schwab, 2016, Chen and Schwab, 2016, Chen et al., 2017]...
  3. transport maps (polynomials, radial basis functions, deep neural networks) [El Moselhy and Marzouk, 2012, Spantini et al., 2018, Rezende and Mohamed, 2015, Liu and Wang, 2016, Detommaso et al., 2018, Chen et al., 2019]...

# Computational methods

- **Surrogate models to reduce the large-scale computation**
  1. polynomial approximation (stochastic spectral, stochastic collocation) [Marzouk et al., 2007, Marzouk and Xiu, 2009, Schwab and Stuart, 2012, Chen et al., 2017, Farcas et al., 2019]...
  2. model reduction (POD, greedy reduced basis) [Wang and Zabaras, 2005, Lieberman et al., 2010, Nguyen et al., 2010, Lassila et al., 2013, Cui et al., 2015, Chen and Schwab, 2016, Chen and Ghattas, 2019]...
  3. multilevel/multifidelity (MCMC, stochastic collocation) [Dodwell et. al., 2015, Teckentrup et. al., 2015, Scheichl et. al., 2017, Peherstorfer. et. al., 2018, Farcas et. al., 2019]...

Aim for this talk:

**Fast** and **scalable** Bayesian inference in **high** dimensions by exploiting intrinsic low-dimensionality in both parameter and state spaces, using

- projected transport map in parameter space
- reduced basis approximation in state space

# Computational methods

- **Surrogate models to reduce the large-scale computation**
  1. polynomial approximation (stochastic spectral, stochastic collocation) [Marzouk et al., 2007, Marzouk and Xiu, 2009, Schwab and Stuart, 2012, Chen et al., 2017, Farcas et al., 2019]...
  2. model reduction (POD, greedy reduced basis) [Wang and Zabaras, 2005, Lieberman et al., 2010, Nguyen et al., 2010, Lassila et al., 2013, Cui et al., 2015, Chen and Schwab, 2016, Chen and Ghattas, 2019]...
  3. multilevel/multifidelity (MCMC, stochastic collocation) [Dodwell et. al., 2015, Teckentrup et. al., 2015, Scheichl et. al., 2017, Peherstorfer. et. al., 2018, Farcas et. al., 2019]...

Aim for this talk:

**Fast** and **scalable** Bayesian inference in **high** dimensions by exploiting intrinsic low-dimensionality in both parameter and state spaces, using

- projected transport map in parameter space
- reduced basis approximation in state space

# Computational methods

- **Surrogate models to reduce the large-scale computation**
  1. polynomial approximation (stochastic spectral, stochastic collocation) [Marzouk et al., 2007, Marzouk and Xiu, 2009, Schwab and Stuart, 2012, Chen et al., 2017, Farcas et al., 2019]...
  2. model reduction (POD, greedy reduced basis) [Wang and Zabaras, 2005, Lieberman et al., 2010, Nguyen et al., 2010, Lassila et al., 2013, Cui et al., 2015, Chen and Schwab, 2016, Chen and Ghattas, 2019]...
  3. multilevel/multifidelity (MCMC, stochastic collocation) [Dodwell et. al., 2015, Teckentrup et. al., 2015, Scheichl et. al., 2017, Peherstorfer. et. al., 2018, Farcas et. al., 2019]...

Aim for this talk:

**Fast** and **scalable** Bayesian inference in **high** dimensions by exploiting intrinsic low-dimensionality in both parameter and state spaces, using

- projected transport map in parameter space
- reduced basis approximation in state space

# Computational methods

- **Surrogate models to reduce the large-scale computation**
  1. polynomial approximation (stochastic spectral, stochastic collocation) [Marzouk et al., 2007, Marzouk and Xiu, 2009, Schwab and Stuart, 2012, Chen et al., 2017, Farcas et al., 2019]...
  2. model reduction (POD, greedy reduced basis) [Wang and Zabaras, 2005, Lieberman et al., 2010, Nguyen et al., 2010, Lassila et al., 2013, Cui et al., 2015, Chen and Schwab, 2016, Chen and Ghattas, 2019]...
  3. multilevel/multifidelity (MCMC, stochastic collocation) [Dodwell et. al., 2015, Teckentrup et. al., 2015, Scheichl et. al., 2017, Peherstorfer. et. al., 2018, Farcas et. al., 2019]...

Aim for this talk:

**Fast** and **scalable** Bayesian inference in **high** dimensions by exploiting intrinsic low-dimensionality in both parameter and state spaces, using

- projected transport map in parameter space
- reduced basis approximation in state space

# Computational methods

- **Surrogate models to reduce the large-scale computation**
  1. polynomial approximation (stochastic spectral, stochastic collocation) [Marzouk et al., 2007, Marzouk and Xiu, 2009, Schwab and Stuart, 2012, Chen et al., 2017, Farcas et al., 2019]...
  2. model reduction (POD, greedy reduced basis) [Wang and Zabaras, 2005, Lieberman et al., 2010, Nguyen et al., 2010, Lassila et al., 2013, Cui et al., 2015, Chen and Schwab, 2016, Chen and Ghattas, 2019]...
  3. multilevel/multifidelity (MCMC, stochastic collocation) [Dodwell et. al., 2015, Teckentrup et. al., 2015, Scheichl et. al., 2017, Peherstorfer. et. al., 2018, Farcas et. al., 2019]...

Aim for this talk:

**Fast** and **scalable** Bayesian inference in **high** dimensions by exploiting intrinsic low-dimensionality in both parameter and state spaces, using

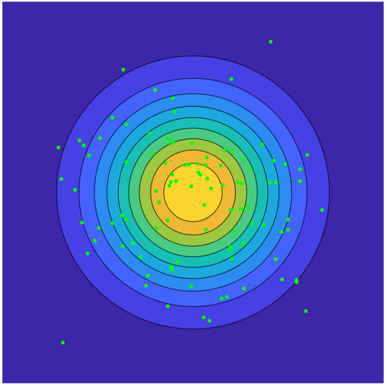
- projected transport map in parameter space
- reduced basis approximation in state space

# Outline

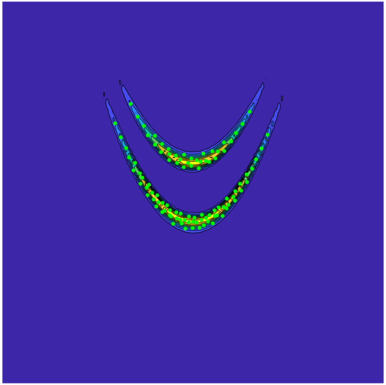
# Transport map

Find a **transport map** $T : \mathbb{R}^d \to \mathbb{R}^d$, such that

$$\theta \sim \pi_0 \to T(\theta) \sim \pi_y,$$



prior



posterior

# Kullback–Leibler divergence

## Definition: **Kullback–Leibler** (KL) divergence

$$\mathcal{D}_{\mathsf{KL}}(\pi_1|\pi_2) = \int_\Theta \pi_1(\theta) \log\left(\frac{\pi_1(\theta)}{\pi_2(\theta)}\right) d\theta.$$

It measures the difference between two probability distribution

$$\mathcal{D}_{\mathsf{KL}}(\pi_1|\pi_2) \geq 0, \text{ and } \mathcal{D}_{\mathsf{KL}}(\pi_1|\pi_2) = 0 \text{ if and only if } \pi_1 = \pi_2, \text{ a.e.}$$

It is not symmetric, thus not a distance

$$\mathcal{D}_{\mathsf{KL}}(\pi_1|\pi_2) \neq \mathcal{D}_{\mathsf{KL}}(\pi_2|\pi_1)$$

Relation to (Shannon) information theory

$$\mathcal{D}_{\mathsf{KL}}(\pi_1|\pi_2) = \underbrace{\mathbb{E}_{\theta\sim\pi_1}[-\log(\pi_2)]}_{\text{cross entropy}} - \underbrace{\mathbb{E}_{\theta\sim\pi_1}[-\log(\pi_1)]}_{\text{entropy}}$$

# Optimization for transport map

- Find a **transport map** $T : \mathbb{R}^d \to \mathbb{R}^d$, such that

$$\theta \sim \pi_0 \to T(\theta) \sim \pi_y,$$

by minimizing the KL divergence

$$\min_{T \in \mathcal{T}} \mathcal{D}_{\mathsf{KL}}(T_\sharp \pi_0 | \pi_y) \quad \Leftrightarrow \quad \min_{T \in \mathcal{T}} \mathcal{D}_{\mathsf{KL}}(\pi_0 | T^\sharp \pi_y).$$

- $T_\sharp$ is a pushforward map satisfying

$$T_\sharp \pi_0(\theta) = \pi_0(T^{-1}(\theta)) |\mathrm{det} \nabla T^{-1}(\theta)|,$$

$T^\sharp$ is a pullback map satisfying

$$T^\sharp \pi_y(\theta) = \pi_y(T(\theta)) |\mathrm{det} \nabla T(\theta)|.$$

- $\mathcal{T}$ is a tensor-product function space $\mathcal{H}^d = \mathcal{H} \otimes \cdots \otimes \mathcal{H}$.

# Composition of transport map

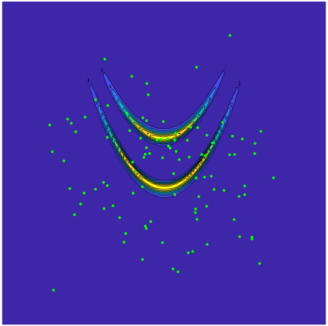
Instead of seeking **one complex (highly nonlinear)** transport map $T$, we look for **composition of a sequence of simple** transport maps

$$T = T_L \circ T_{L-1} \circ \cdots \circ T_1 \circ T_0, \quad L \in \mathbb{N},$$

- **perturbation of identity**:

$$T_l(\theta) = I(\theta) + Q_l(\theta),$$

- identity map $I(\theta) = \theta$
- perturbation map $Q_l : \mathbb{R}^d \to \mathbb{R}^d$



# Optimization of each transport map

At each $l = 0, 1, \dots,$ we define

$$\pi_{l+1} := (T_l \circ \cdots \circ T_0)_\sharp \pi_0 \Longleftrightarrow \pi_{l+1} = (T_l)_\sharp \pi_l$$

We introduce a cost functional

$$\mathcal{J}_l[Q] := \mathcal{D}_{\mathsf{KL}}((I + Q)_\sharp \pi_l | \pi_y). \tag{1}$$

One step optimization of $\mathcal{J}_l(Q)$ w.r.t. $Q$ leads to

$$T_l = I + \alpha_l Q_l,$$

with step size $\alpha_l > 0$ (learning rate, line search).

## Optimization methods

- Gradient descent method: steepest descent [Liu and Wang, 2016]

$$Q_l = -D\mathcal{J}_l[\mathbf{0}].$$

- Newton method: solve the linear system [Detommaso et al., 2018]

$$D^2\mathcal{J}_l[\mathbf{0}](V, Q_l) = -D\mathcal{J}_l[\mathbf{0}](V), \quad \forall\, V \in \mathcal{T}.$$

# Optimization of each transport map

## Calculus of variation

- The first variation $D\mathcal{J}_l[\mathbf{0}]$ at $Q = \mathbf{0}$ in direction $V \in \mathcal{T}$

$$D\mathcal{J}_l[\mathbf{0}](V) := -\mathbb{E}_{\pi_l}\left[(V(\theta))^\top \nabla_\theta \log(\pi_y(\theta)) + \text{trace}(\nabla_\theta V(\theta))\right]$$

- The second variation $D^2\mathcal{J}_l[\mathbf{0}]$ at $Q = \mathbf{0}$ in directions $V, W \in \mathcal{T}$

$$D^2\mathcal{J}_l[\mathbf{0}](V,W) := -\mathbb{E}_{\pi_l}\left[(V(\theta))^\top \nabla^2_\theta \log(\pi_y(\theta))\ W(\theta) - \text{trace}(\nabla_\theta W(\theta)\ \nabla_\theta V(\theta))\right]$$

Recall the Bayes' rule:

$$\underbrace{\pi_y(\theta)}_{\text{posterior}} = \frac{1}{\pi(y)} \underbrace{\pi(y|\theta)}_{\text{likelihood}} \underbrace{\pi_0(\theta)}_{\text{prior}},$$

which leads to

$$\nabla_\theta \log(\pi_y(\theta)) = \frac{\nabla_\theta \pi_y(\theta)}{\pi_y(\theta)} = \frac{\nabla_\theta(\pi(y|\theta)\pi_0(\theta))}{\pi(y|\theta)\pi_0(\theta)}$$

Key observation: the intractable model evidence $\pi(y)$ is canceled out.

# Reproducing Kernel Hilbert Space (RKHS)

$\mathcal{T}$ is a tensor-product function space $\mathcal{H}^d = \mathcal{H} \otimes \cdots \otimes \mathcal{H}$.

- tensor-product polynomials [El Moselhy and Marzouk, 2012, Spantini et al., 2018],
- radial basis/kernel functions [Liu and Wang, 2016, Detommaso et al., 2018].

## Reproducing Kernel Hilbert Space $\mathcal{H}$

There exists a function $k_\theta \in \mathcal{H}$ for every $\theta \in \Theta$, such that

$$v(\theta) = \langle v, k_\theta \rangle \quad \forall v \in \mathcal{H},$$

which implies existence of $k_{\theta'} \in \mathcal{H}$ for every $\theta' \in \Theta$ such that

$$k_{\theta'}(\theta) = \langle k_{\theta'}, k_\theta \rangle =: k(\theta, \theta') \quad \text{reproducing kernel}$$

Many choices: bilinear, polynomials, Bergman, radial basis functions

# $N$-dimensional approximation of RKHS

## Gaussian kernel

$$k(\theta, \theta') = \exp\left(-\frac{1}{2h}(\theta - \theta')^\top \mathbb{M}(\theta - \theta')\right).$$

To account for the geometry of the posterior, [Detommaso et al., 2018]

$$\mathbb{M} = \bar{\mathbb{H}} := \mathbb{E}_{\pi_l}\left[-\nabla_\theta^2 \log(\pi_y(\theta))\right], \; h = d, \qquad \text{v.s.} \qquad \mathbb{M} = \mathbb{I} \in \mathbb{R}^{d\times d}.$$

**Finite dimensional approximation of RKHS**:

$$\mathcal{H}_N^l = \text{span}(k_1^l(\theta), \dots, k_N^l(\theta)) \subset \mathcal{H},$$

where the basis functions are taken as

$$k_n^l(\theta) = k(\theta, \theta_n^l), \quad n = 1, \dots, N,$$

where $\theta_n^l \sim \pi_l$ are particles transported from $\theta_n^0 \sim \pi_0$ by

$$\theta_n^l = (T_l \circ \cdots \circ T_0)(\theta_n^0), \quad n = 1, \dots, N.$$

# Stein variational gradient descent (SVGD)

[Liu and Wang, 2016]

- For $D\mathcal{J}_l[\mathbf{0}](V) = \langle D\mathcal{J}_l[\mathbf{0}], V\rangle_{\mathcal{H}^d}$, by the reproducing property

$$\langle D\mathcal{J}_l[\mathbf{0}], V\rangle_{\mathcal{H}^d} = -\langle \mathbb{E}_{\pi_l}[\nabla_\theta \log(\pi_y(\theta))k(\theta,\theta') + \nabla_\theta k(\theta,\theta')], V(\theta')\rangle.$$

- For gradient descent, we have (by notation $k_n^l(\theta) = k(\theta, \theta_n^l)$)

$$Q_l(\theta_n^l) = -D\mathcal{J}_l[\mathbf{0}](\theta_n^l) = \mathbb{E}_{\pi_l}\left[\nabla_\theta \log(\pi_y(\theta))k_n^l(\theta) + \nabla_\theta k_n^l(\theta)\right]$$

- Sample average approximation (SAA): $\theta_m^l \sim \pi_l$, $m = 1, \ldots, N$

$$Q_l(\theta_n^l) \approx \frac{1}{N}\sum_{m=1}^{N} \nabla_\theta \log(\pi_y(\theta_m^l))k_n^l(\theta_m^l) + \nabla_\theta k_n^l(\theta_m^l).$$

- Particle updates by the transport map

$$\theta_n^{l+1} = T_l(\theta_n^l) := \theta_n^l + \alpha_l\, Q_l(\theta_n^l), \quad n = 1, \ldots, N.$$

# Stein variational Newton (SVN) [Detommaso et al., 2018]

- We seek $Q_l \in \mathcal{T}_N^l = (\mathcal{H}_N^l)^d$, where $\mathcal{H}_N^l = \text{span}(k_1^l(\theta), \dots, k_N^l(\theta))$,

$$Q_l(\theta) = \sum_{n=1}^{N} c_n^l k_n^l(\theta),$$

where the coefficients $c_n^l \in \mathbb{R}^d$, with $\boldsymbol{c}^l = (c_1^l, \dots, c_N^l) \in \mathbb{R}^{dN}$.

- For the Newton system: find $Q_l \in \mathcal{T}_N^l$ such that

$$D^2\mathcal{J}_l[\mathbf{0}](V, Q_l) = -D\mathcal{J}_l[\mathbf{0}](V), \quad \forall\, V \in \mathcal{T}_N^l,$$

which, by using the reproducing property, becomes

$$\mathbb{H}\boldsymbol{c}^l = -\boldsymbol{g}^l,$$

gradient: $\boldsymbol{g}^l = (g_1^l, \dots, g_N^l) \in \mathbb{R}^{dN}$, Hessian: $\mathbb{H} \in \mathbb{R}^{dN \times dN}$.

# Stein variational Newton (SVN) [Detommaso et al., 2018]

- The gradient $\boldsymbol{g}^l = (g_1^l, \dots, g_N^l) \in \mathbb{R}^{dN}$, with $g_m^l \in \mathbb{R}^d$ given by

$$g_m^l = -\frac{1}{N}\sum_{i=1}^{N} \nabla_\theta \log(\pi_y(\theta_i^l)) k_m^l(\theta_i^l) + \nabla_\theta k_m^l(\theta_i^l)$$

- The Hessian $\mathbb{H} \in \mathbb{R}^{dN\times dN}$: with $\mathbb{H}_{mn} \in \mathbb{R}^{d\times d}$ given by

$$\mathbb{H}_{mn} = \frac{1}{N}\sum_{i=1}^{N} -\nabla_\theta^2 \log(\pi_y(\theta_i^l)) k_m^l(\theta_i^l) k_n^l(\theta_i^l) + \nabla_\theta k_m^l(\theta_i^l)(\nabla_\theta k_n^l(\theta_i^l))^\top.$$

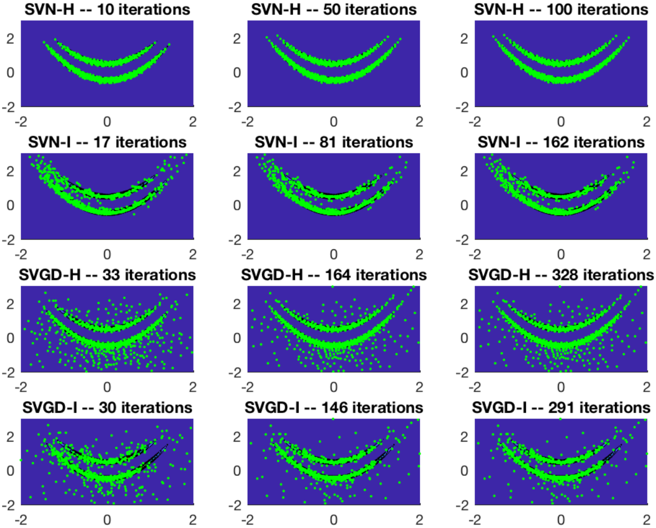
- Decouple $dN \times dN$ system to $N$ systems of size $d \times d$

$$\bar{\mathbb{H}}_m c_m^l = -g_m^l, \quad m = 1, \dots, N,$$

with diagonal approximation

$$\bar{\mathbb{H}}_m = \frac{1}{N}\sum_{i=1}^{N} -\nabla_\theta^2 \log(\pi_y(\theta_i^l)) k_m^l(\theta_i^l) k_m^l(\theta_i^l) + \nabla_\theta k_m^l(\theta_i^l)(\nabla_\theta k_m^l(\theta_i^l))^\top.$$

# SVGD vs SVN with $\mathbb{M} = \mathbb{I}$ vs $\mathbb{M} = \bar{\mathbb{H}}$



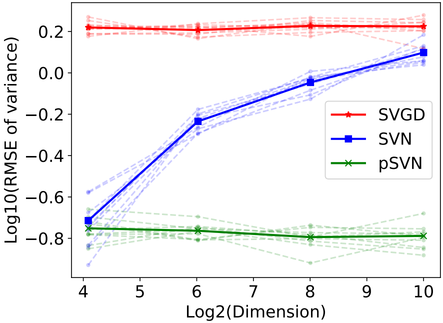

G. Detommaso, T. Cui, Y. Marzouk, A. Spantini, R. Scheichl. A Stein variational Newton method. NeurIPS, 2018.

# Outline

# Computational challenges in high dimensions

## Curse of dimensionality: $d \gg 1$

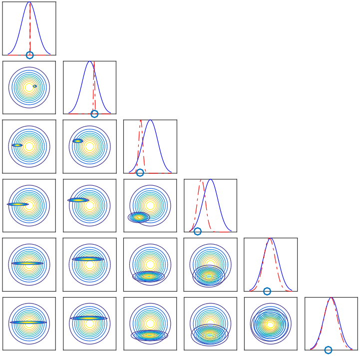
The number $N$ of basis functions grows rapidly (exponentially) w.r.t. the dimension $d$ to achieve map representation with required accuracy.

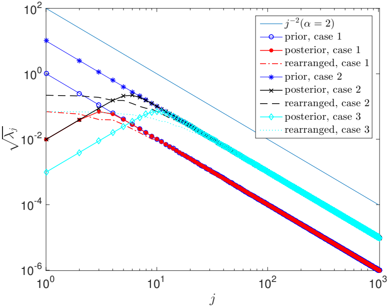



P. Chen, K. Wu, J. Chen, T. O'Leary-Roseberry, O. Ghattas. Projected Stein variational Newton: A fast and scalable Bayesian inference method in high dimensions. NeurIPS, 2019.

The posterior $\neq$ the prior in a low-dimensional subspace.

- high correlation in different dimensions;
- forward map is smoothing/regularizing;
- parameters are anistropic, e.g., KL expansion.






P. Chen, U. Villa, O. Ghattas. Hessian-based adaptive sparse quadrature for infinite-dimensional Bayesian inverse problems. CMAME, 2017.

# Projection [Constantine et. al. 2014, Cui et. al. 2014]

- We denote a basis of the subspace of dimension $r \ll d$ as

$$\Psi = (\psi_1, \ldots, \psi_r) \in \mathbb{R}^{d \times r}.$$

- We project $\theta$ to the low-dimensional subspace as

$$\theta^r = \sum_{i=1}^{r} \psi_i \psi_i^\top \theta = \Psi w.$$

- As a result, we consider the projected posterior

$$\pi_y^r(\theta) = \frac{1}{\pi^r(y)} \pi(y|\theta^r) \pi_0(\theta), \tag{2}$$

where the maginal density

$$\pi^r(y) = \mathbb{E}_{\pi_0}[\pi(y|\theta^r)].$$

# Projected Stein variational methods

- By decomposition $\theta = \theta^r + \theta^\perp$, we have

$$\pi_y^r(\theta) = \pi(y|\theta^r)\ \pi_0(\theta^r)\ \pi_0(\theta^\perp|\theta^r).$$

- With $\theta^\perp$ frozen, by $\theta^r = \Psi w$, we define

$$p_0(w) := \pi_0(\theta^r), \quad p_y(w) := \pi_y^r(\theta^r) = \pi(y|\theta^r)\pi_0(\theta^r).$$

- We seek $T = T_L \circ T_{L-1} \circ \cdots \circ T_1 \circ T_0 : \mathbb{R}^r \to \mathbb{R}^r$, such that

$$\min_{T \in \mathcal{T}} D_{KL}(T_\sharp p_0 | p_y).$$

- Apply **SVGD/SVN** in $\mathbb{R}^r$ for $w$, **pSVGD/pSVN** where

$$\nabla_w \log(p_y(w)) = \Psi^\top \nabla_\theta \log(\pi_y^r(\theta^r)),$$

and

$$\nabla_w^2 \log(p_y(w)) = \Psi^\top \nabla_\theta^2 \log(\pi_y^r(\theta^r))\Psi.$$

# Basis construction

The basis functions $\Psi$ for projection are obtained by

$$H\psi_i = \lambda_i C_0^{-1}\psi_i, \quad i = 1, \ldots, r,$$

which corresponds to the $r$ largest (in $|\cdot|$) eigenvalues, i.e., $|\lambda_1| \geq \cdots \geq |\lambda_r|$. $C_0$: prior covariance. With $\eta_y(\theta) = -\log(\pi(y|\theta))$

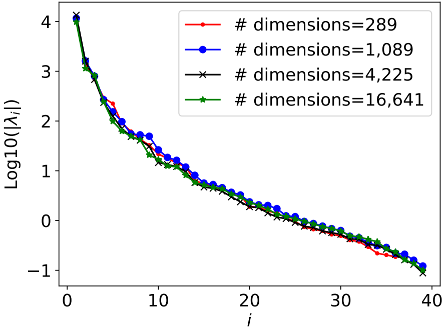
- **Gradient-based subspace:**

$$H = \mathbb{E}_\pi\left[\nabla_\theta \eta_y(\theta)(\nabla_\theta \eta_y(\theta))^\top\right].$$

- **Hessian-based subspace:**

$$H = \mathbb{E}_\pi\left[\nabla_\theta^2 \eta_y(\theta)\right].$$

- **Choice of the density $\pi$:** density at step $l$, i.e., $\pi_l$.



**Algorithm 1** pSVN in parallel using MPI

**Input**: $N$ prior samples, $\theta_1^0, \dots, \theta_N^0$, in each of $K$ cores.
**Output**: posterior samples $\theta_1^y, \dots, \theta_N^y$ in each core.
Perform projection to get $\theta_n = \theta_n^r + \theta_n^\perp$ and the samples $w_n^{l-1}$.
*Perform MPI_Allgather for $w_n^{l-1}$, $n = 1, \dots, M$.*
**repeat**
    Compute the gradient and Hessian.
    *Perform MPI_Allgather for the gradient and Hessian.*
    Compute the kernel and its gradient.
    Assemble and solve Newton system for $c_1, \dots, c_N$.
    Perform a line search to get $w_1^l, \dots, w_N^l$.
    *Perform MPI_Allgather for $w_n^l$, $n = 1, \dots, N$.*
    Update the samples $\theta_n^r = \Psi w_n^l + \bar{\theta}$, $n = 1, \dots, N$.
    Set $l \leftarrow l + 1$.
**until** A stopping criterion is met.
Reconstruct samples $\theta_n^y = \theta_n^r + \theta_n^\perp$, $n = 1, \dots, N$.

**Algorithm 2** Adaptive pSVN

1: **Input**: $N$ prior samples, $\theta_1^0, \dots, \theta_N^0$, in each of $K$ cores.
2: **Output**: posterior samples $\theta_1^y, \dots, \theta_N^y$ in each core.
3: Set level $l_2 = 1$, $\theta_n^{l_2-1} = \theta_n^0$, $n = 1, \dots, N$.
4: **repeat**
5: Perform the eigendecomposition and form the bases $\Psi^{l_2}$.
6: Apply **Algorithm** pSVN to update the samples $[\theta_1^{l_2}, \dots, \theta_N^{l_2}] = \text{pSVN}([\theta_1^{l_2-1}, \dots, \theta_N^{l_2-1}], K, \Psi^{l_2})$.
7: Set $l_2 \leftarrow l_2 + 1$.
8: **until** A stopping criterion is met.

## Advantages:

- Avoids/alleviates the curse of dimensionality.
- Largely reduces computational cost with $r \ll d$.
- Converges faster in low-dimensional space.
- Parallel computation with reduced communication.

# Projection error estimates

## Assumption

*Assume that the parameter-to-observable map $\mathcal{O}$ satisfies:*

1. *There exists a constant $C_{\mathcal{O}} > 0$ such that $\mathbb{E}_{\pi_0}[||\mathcal{O}(\cdot)||_\Gamma] \leq C_{\mathcal{O}}$.*
2. *For every $b > 0$, there exists a constant $C_b > 0$ such that*

$$||\mathcal{O}(\theta_1) - \mathcal{O}(\theta_2)||_\Gamma \leq C_b||\theta_1 - \theta_2||_\Theta, \quad \textit{for } \max\{||\theta_1||_X, ||\theta_2||_\Theta\} < b.$$

## Theorem

*Under Assumption 1, for Hessian-based projection, we have*

$$\mathcal{D}_{KL}(\pi_y \,|\, \pi_y^r) \leq C||\theta - \theta^r||_\Theta,$$

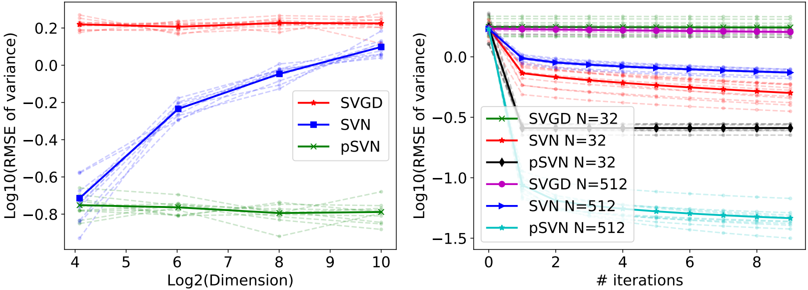
*$C$ independent of $r$. For gradient-based projection, based on a result in [Zahm et. al., 2018], we obtain (with $C$ independent of $r$)*

$$\mathcal{D}_{KL}(\pi_y \,|\, \pi_y^r) \leq C \sum_{i=r+1}^{d} \lambda_i.$$

We first consider a linear parameter-to-observable map

$$\mathcal{O}(\theta) = A\theta, \quad A = O(B\theta), \quad B = (L+M)^{-1},$$

where $L$ and $M$ are the discrete Laplacian and mass matrices in the PDE model $-\triangle u + u = \theta$, in $(0,1)$, $u(0) = 0$, $u(1) = 1$. Gaussian prior $\mathcal{N}(0, C_0)$, $C_0$ is discretized from $(I - 0.1\triangle)^{-1}$ with Laplace operator $\triangle$.



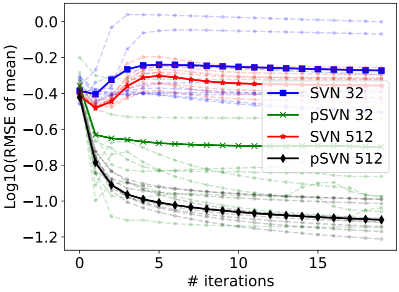

Decay of the RMSE of the L2 of pointwise variance of the parameter w.r.t. dimension $d = 16, 64, 256, 1024$ with $N = 128$ samples (left), and with $N = 32$, and $512$ samples in parameter dimension $d = 256$ w.r.t. # iterations (right).

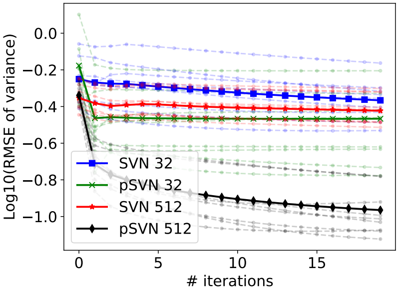
# Numerical results: Accuracy

We consider a nonlinear Bayesian inverse problem with

$$\mathcal{O}(\theta) = O(S(\theta)), \quad u = S(\theta), \quad -\nabla \cdot (e^{\theta}\nabla u) = 0, \text{ in } (0,1)^2$$

Gaussian prior $\mathcal{N}(0, C_0)$, where $C_0$ is a discretization of $(I - 0.1\triangle)^{-2}$. We test the accuracy against a dimension-independent likelihood informed (DILI) MCMC method with 10,000 samples as reference.



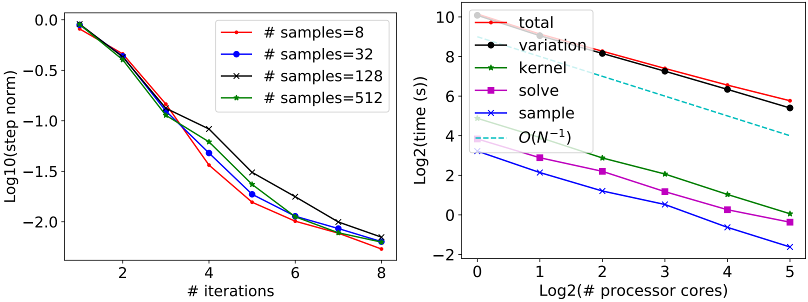

Decay of the RMSE of the L2 of the mean (left) and pointwise variance (right) of the parameter with dimension $d = 1089$ and $N = 32$ and $512$ samples.

# Numerical results: Scalability

We consider a nonlinear Bayesian inverse problem with

$$\mathcal{O}(\theta) = O(S(\theta)), \quad u = S(\theta), \quad -\nabla \cdot (e^{\theta} \nabla u) = 0, \text{ in } (0,1)^2$$

Gaussian prior $\mathcal{N}(0, C_0)$, where $C_0$ is a discretization of $(I - 0.1\triangle)^{-2}$. We test the accuracy against a dimension-independent likelihood informed (DILI) MCMC method with 10,000 samples as reference.




Left: Decay of the averaged norm of the update $w^{l+1} - w^l$ w.r.t. the iteration number $l$, with increasing number of samples. Right: Decay of the wall clock time of different computational components w.r.t. increasing # cores.

# Summary

**Take away message**:

- SVN provides good samples for complex posterior.
- pSVN is scalable to address the curse-of-dimensionality.

**Ongoing**:

- Bayesian optimal experimental design with Keyi Wu.
- Triangular map and data assimilation with Joshua Chen.
- Deep learning for transport map with Tom O'Leary-Roseberry.
- Gravitational wave inversion with Bassel Saleh, Alex Leviyev.
- Integration with model reduction with Zihang Zhang.
- Convergence analysis w.r.t. # particles, parameter dimensions.
- Multilevel parallel implementation w.r.t. particles and PDE solves.

# Outline

# PDE-constrained Bayesian inversion

- We have the data model

$$y = \mathcal{B}(u(\theta)) + \xi$$

where $u$ is the solution of the PDE (in weak form)

$$A(u(\theta), v; \theta) = F(v) \quad v \in V$$

$\mathcal{B} : V \to Y$ is a vector of observational functionals.

- Examples: linear diffusion, elasiticity, Stokes flow, acoustic, etc.,

$$-\nabla \cdot (\kappa(\theta)\nabla u) = f, \qquad \text{in } D,$$

with suitable boundary conditions, which leads to

$$A(u, v; \theta) = \int_D \kappa(x, \theta)\nabla u(x, \theta) \cdot \nabla v(x) dx, \; F(v) = \int_D f(x)v(x)dx.$$

- With Gaussian noise $\xi \in \mathcal{N}(0, \Gamma)$, we define the potential

$$\eta_y(\theta) := \frac{1}{2}(y - \mathcal{B}(u(\theta)))^T \Gamma^{-1}(y - \mathcal{B}(u(\theta))) \Rightarrow \pi(y|\theta) = \log(-\eta_y(\theta)).$$

# High-fidelity approximation of the potential $\eta_y$

E.g. finite element, we consider: find $u_h \in V_h \subset V$ such that

$$A(u_h, v_h, \theta) = F(v_h) \quad \forall v_h \in V_h. \tag{3}$$

Then the data model is given by

$$y = \mathcal{B}(u_h(\theta)) + \xi,$$

then for $\xi \sim \mathcal{N}(0, \Gamma)$ the likelihood function is given by

$$\pi(y|\theta) = \exp(-\eta_y(u_h(\theta))),$$

where the potential $\eta_y(u_h(\theta))$ (nonlinear w.r.t. $u_h$)

$$\eta_y(u_h(\theta)) = \frac{1}{2}(y - \mathcal{B}(u_h(\theta)))^T \Gamma^{-1} (y - \mathcal{B}(u_h(\theta))).$$

For SVGD, and the projected SVGD, we also need

$$-\nabla_\theta \log(\pi_y(\theta)) = \nabla_\theta \eta_y(u_h(\theta)) + \frac{\nabla_\theta \pi_0(\theta)}{\pi_0(\theta)}.$$

# High-fidelity approximation of the gradient $\nabla_\theta \eta_y$

We form a Lagrangian

$$L(u_h, p_h, \theta) = \eta_y(u_h) + A(u_h, p_h, \theta) - F(p_h),$$

$\partial_v L w_h = 0$ to obtain the adjoint $p_h$, i.e., find $p_h \in V$ such that

$$A(w_h, p_h; \theta) = -\partial_u \eta_y|_{u_h}(w_h) \quad \forall w_h \in V_h, \tag{4}$$

where

$$\partial_u \eta_y|_{u_h}(w_h) = -\mathcal{B}(w_h)^T \Gamma^{-1}(y - \mathcal{B}(u_h)).$$

Then the gradient is given by

$$\nabla_\theta \eta_y(u_h(\theta)) = \partial_\theta L(u_h, p_h; \theta) = \partial_\theta A(u_h, p_h, \theta).$$

# Model reduction

## High-fidelity approximation

Finite element space $V_h$,

$$\dim(V_h) = N_h$$

Given $\theta$, find $u_h \in V_h$ s.t.

$$A(u_h, v_h; \theta) = F(v_h) \ \forall v_h \in V_h$$

The algebraic system is

$$\mathbb{A}_h(\theta)\boldsymbol{u}_h = \boldsymbol{f}_h$$

$$\mathbb{V} = [\boldsymbol{\psi}_1, \dots, \boldsymbol{\psi}_N]$$

$$\mathbb{V}^T\boldsymbol{u}_h = \boldsymbol{u}_N$$

$$\mathbb{V}^T\mathbb{A}_h(\theta)\mathbb{V} = \mathbb{A}_N(\theta)$$

$$\mathbb{V}^T\boldsymbol{f}_h = \boldsymbol{f}_N$$

## Reduced basis approximation

Reduced basis space $V_N \subset V_h$,

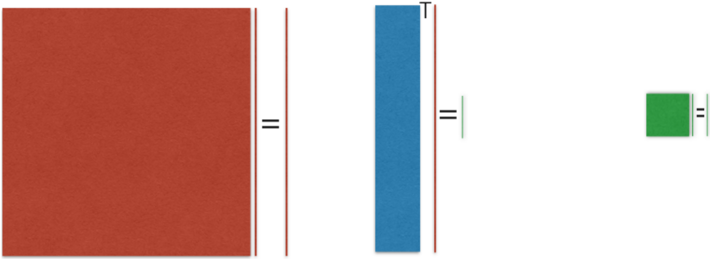
$$\dim(V_N) = N$$

Given $\theta$, find $u_N \in V_N$ s.t.

$$A(u_N, v_N; \theta) = F(v_N) \ \forall v_N \in V_N$$

The algebraic system is

$$\mathbb{A}_N(\theta)\boldsymbol{u}_N = \boldsymbol{f}_N$$



# Model reduction: Building blocks

## POD/SVD

Training samples

$$\Xi_t = \{\theta^n, n = 1, \ldots, N_t\}$$

Compute snapshots

$$\mathbb{U} = [\boldsymbol{u}_h(\theta^1), \ldots, \boldsymbol{u}_h(\theta^{N_t})]$$

Perform SVD

$$\mathbb{U} = \mathbb{V}\Sigma\mathbb{W}^T$$

Extract bases $\mathbb{V}[1:N,:]$

$$N = \text{argmin}_n \, \mathcal{E}_n(\Sigma) \geq 1 - \varepsilon$$

## Greedy algorithm

Training samples

$$\Xi_t = \{\theta^n, n = 1, \ldots, N_t\}$$

Initialize $V_N$ for $N = 1$ as

$$V_N = \text{span}\{u_h(\theta^1)\}$$

Pick next sample such that

$$\theta^{N+1} = \text{argmax}_{\theta\in\Xi_t} \, \Delta_N(\theta)$$

Update bases $V_{N+1}$ as

$$V_N \oplus \text{span}\{u_h(\theta^{N+1})\}$$

## Offline-Online

Affine assumption/approx.

$$A = \sum_{q=1}^{Q} \theta_q(\theta) A_q$$

Offline computation once

$$\mathbb{A}_N^q = \mathbb{V}^T \mathbb{A}_h^q \mathbb{V}$$

Online assemble

$$\mathbb{A}_N(\theta) = \sum_{q=1}^{Q} \theta_q(\theta) \mathbb{A}_N^q$$

Online solve

$$\mathbb{A}_N(\theta)\boldsymbol{u}_N = \boldsymbol{f}_N$$

## Goal-oriented a-posteriori error estimate $\Delta_N(\theta)$ – dual weighted residual

$$\Delta_N(\theta) = A(u_N, p_N, \theta) - F(p_N)$$

# Reduced basis approximation of the potential $\eta_y$

RB approximation for the adjoint problem: find $p_N \in W_N$ s.t.

$$A(w_N, p_N, \theta) = -\partial_u \eta_y|_{u_N}(w_N) \quad \forall w_N \in W_N.$$

The goal-oriented a-posterior error estimate is given by

$$\Delta_N(\theta) = A(u_N, p_N, \theta) - F(p_N).$$

RB approximation for the potential $\eta_y(\theta)$:

$$\eta_{y,N}(\theta) = \eta_y(u_N(\theta)).$$

Dual-weighted residual correction:

$$\eta^{\Delta}_{y,N}(\theta) = \eta_{y,N}(\theta) + \Delta_N(\theta).$$

# Reduced basis approximation of the gradient $\nabla_\theta \eta_y$

With the RB state $u_N$ and adjoint $p_N$, the gradient is given by

$$\nabla_\theta \eta_y(u_N(\theta)) = \partial_\theta A(u_N, p_N; \theta).$$

For the modified potential $\eta_{y,N}^{\Delta}(\theta)$, we form the Lagrangian

$$\begin{aligned} L(u_N, p_N, \hat{u}_N, \hat{p}_N; \theta) = \eta_{y,N}^{\Delta}(\theta) + A(u_N, \hat{u}_N; \theta) - F(\hat{u}_N) \\ + A(\hat{p}_N, p_N; \theta) + \nabla_u \eta_y|_{u_N}(\hat{p}_N), \end{aligned}$$

and solve the variational problem: find $\hat{p}_N \in W_N$

$$A(\hat{p}_N, w_N; \theta) = F(w_N) - A(u_N, w_N; \theta), \quad \forall w_N \in W_N,$$

and the variational problem: find $\hat{u}_N \in V_N$

$$A(v_N, \hat{u}_N; \theta) = -A(v_N, p_N; \theta) - \partial_u \eta_y|_{u_N}(v_N) - \nabla_u^2 \eta_y|_{u_N}(\hat{p}_N, v_N), \ \forall v_N \in V_N,$$

which leads to the gradient

$$\nabla_\theta \eta_{y,N}^{\Delta}(\theta) = \partial_\theta L(u_N, p_N, \hat{u}_N, \hat{p}_N; \theta).$$

# Error estimates for the state and adjoint

## Assumption: Well-posedness

The bilinear form $A(\cdot, \cdot; \theta) : V \times V \to \mathbb{R}$ and linear form $F(\cdot) : V \to \mathbb{R}$ satisfy

A1 At any $\theta \in \Theta$, there exist a coercivity constant $\alpha(\theta) > 0$ and a continuity constant $\gamma(\theta) > 0$ such that

$$\alpha(\theta)||w||_V^2 \leq A(w, w; \theta) \text{ and } A(w, v; \theta) \leq \gamma(\theta)||w||_V||v||_V, \ \forall w, v \in V.$$

The linear functional $F(\cdot) : V \to \mathbb{R}$ is bounded with norm

$$||F(\cdot)||_{V'} < \infty.$$

A2 Moreover, $A(\cdot, \cdot; \theta)$ is continuously differentiable w.r.t. $\theta$ at every $\theta \in \Theta$, and for each $j = 1, \dots, d$, there exists $\rho_j(\theta) < \infty$ such that

$$\partial_{\theta_j} A(w, v; \theta) \leq \rho_j(\theta)||w||_V||v||_V, \ \forall w, v \in V.$$

# Error estimates for the state $u$ and adjoint $p$

Let $e_r^u(\theta)$ and $e_r^p(\theta)$ denote the RB state and adjoint errors

$$e_r^u(\theta) := u_h(\theta) - u_N(\theta), \quad e_r^p(\theta) := p_h(\theta) - p_N(\theta).$$

Let $R_u(u_N, \cdot; \theta)$ denotes the residual of the state equation

$$R_u(u_N, v_h; \theta) = A(u_N, v_h; \theta) - F(v_h; \theta) \quad \forall v_h \in V_h,$$

and $R_p(p_N, \cdot; \theta)$ denotes the residual of the adjoint equation

$$R_p(w_h, p_N; \theta) = A(w_h, p_N; \theta) + \nabla_u \eta_y|_{u_N}(w_h) \quad \forall w_h \in V_h.$$

## Lemma: Error estimates for the state $u$ and adjoint $p$

Under the well-posedness assumption, for any $\theta \in \Theta$, there holds

$$||e_r^u(\theta)||_V \leq \frac{1}{\alpha(\theta)} ||R_u(u_N, \cdot; \theta)||_{V'},$$

and

$$||e_r^p(\theta)||_V \leq \frac{1}{\alpha(\theta)} ||R_p(\cdot, p_N; \theta)||_{V'} + \frac{C_{\mathcal{O}}}{\alpha(\theta)} ||e_r^u(\theta)||_V.$$

# Error estimates for the potential $\eta_y$ and gradient $\nabla_\theta \eta_y$

## Lemma: Error esstimates for $\eta_{y,N}(\theta)$ and $\eta^{\Delta}_{y,N}(\theta)$.

There exists constant $C(\theta) > 0$ for each $\theta \in \Theta$, independent of $N$, s.t.

$$|e^{\eta}_r(\theta)| := |\eta_y(\theta) - \eta_{y,N}(\theta)| \leq C(\theta)||e^u_r(\theta)||_V.$$

There exists constant $C_1(\theta) > 0$ for each $\theta \in \Theta$, independent of $N$, s.t.

$$|e^{\Delta}_r(\theta)| := |\eta_y(\theta) - \eta^{\Delta}_{y,N}(\theta)| \leq C||e^u_r(\theta)||_V(||e^u_r(\theta)||_V + ||e^p_r(\theta)||_V).$$

## Lemma: Error esstimates for $\nabla_\theta \eta_{y,N}(\theta)$ and $\nabla_\theta \eta^{\Delta}_{y,N}(\theta)$

There exist $C_1(\theta), C_2(\theta) > 0$ for each $\theta \in \Theta$, independent of $N$, s.t.

$$||\nabla_\theta e^{\eta}_r(\theta)||_1 \leq C_1(\theta)||\nabla_\theta e^u_r(\theta)||_{V^d} + C_2(\theta)||\nabla_\theta u_N(\theta)||_{V^d}||e^u_r(\theta)||_V.$$

There exist $C_1(\theta), , C_2(\theta), C_3(\theta), C_4(\theta) > 0$, independent of $N$, such that

$$\begin{aligned}||\nabla_\theta e^{\Delta}_r(\theta)||_1 \leq\ & C_1||\nabla_\theta e^u_r(\theta)||_{V^d}||e^p_r(\theta)||_V + C_2||\nabla_\theta e^p_r(\theta)||_{V^d}||e^u_r(\theta)||_V \\ & + C_3||e^u_r(\theta)||_V||e^p_r(\theta)||_V + C_4||\nabla_\theta e^u_r(\theta)||_{V^d}||e^u_r(\theta)||_V.\end{aligned}$$

# Error estimates for the posterior $\pi_y$

## Theorem: Error estimates for the posterior $\pi_y$

$$D_{\mathsf{KL}}(\pi_y^h|\pi_y^r) \leq \mathbb{E}_{\pi_y^h}\left[|e_r^\eta|\right] + \mathbb{E}_{\pi_y^h}[|\exp(e_r^\eta) - 1|],$$

and

$$D_{\mathsf{KL}}(\pi_y^h|\pi_y^\Delta) \leq \mathbb{E}_{\pi_y^h}\left[|e_r^\Delta|\right] + \mathbb{E}_{\pi_y^h}[|\exp(e_r^\Delta) - 1|].$$

## Corollary: Error estimates for the posterior $\pi_y$

Let $\Theta_1 =: \{\theta \in \Theta : e_r^\eta(\theta) < 1\}$, if

$$\mathbb{E}_{\pi_y^h(\Theta\setminus\Theta_1)}[|\exp(e_r^\eta) - 1|] < \delta\mathbb{E}_{\pi_y^h}\left[|e_r^\eta|\right]$$

for some constant $\delta > 0$, we have

$$D_{\mathsf{KL}}(\pi_y^h|\pi_y^r) \leq (3+\delta)\mathbb{E}_{\pi_y^h}\left[|e_r^\eta|\right].$$

The same holds for $D_{\mathsf{KL}}(\pi_y^h|\pi_y^\Delta) \leq (3+\delta)\mathbb{E}_{\pi_y^h}\left[|e_r^\Delta|\right]$.

**Algorithm 3** Adaptive greedy algorithm with Stein samples

**Input:** samples $\theta_m^0 \sim \pi_0$, $m = 1, \dots, M$, tolerance $\varepsilon_r^0$, update step $k$.
**Output:** Stein samples $\theta_m$, $m = 1, \dots, M$.
Initialization: at $\theta = \theta_1^0$, solve the high-fidelity state and adjoint problems for $u_h$ and $p_h$, set $V_r = \text{span}\{u_h\}$ and $W_r = \text{span}\{p_h\}$, compute the reduced matrices and vectors for once.
**while** at step $l = 0, k, 2k, \dots,$ of the SVGD algorithm **do**
  Compute the error indicator $\triangle_N(\theta_m^l)$ for $m = 1, \dots, M$.
  **while** $\max_{m=1,\dots,M} |\triangle_N(\theta_m^l)| > \varepsilon_r^l$ **do**
    Choose $\theta = \text{argmax}_{\theta_m^l, m=1,\dots,M} |\triangle_N(x_m^l)|$.
    Solve the high-fidelity problems for $u_h$ and $p_h$ at $\theta$.
    Enrich the spaces $V_r = V_r \bigoplus \text{span}\{u_h\}$, $W_r = W_r \bigoplus \text{span}\{p_h\}$.
    Compute all the reduced matrices and vectors for once.
    Compute the error indicator $\triangle_N(\theta_m^l)$ for $m = 1, \dots, M$.
  **end while**
  Perform SVGD update with RB approximations.
  Update the tolerance $\varepsilon_r^l$ according to gradient in SVGD algorithm.
**end while**

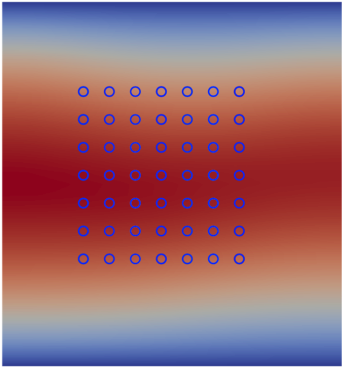
# Numerical example

We consider the diffusion problem

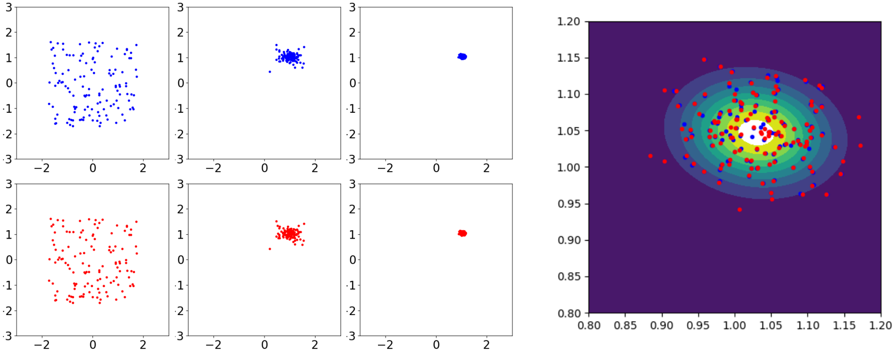
$$-\nabla \cdot (a(\theta, x)\nabla u) = f(x), \quad x \in D = (0,1)^2,$$

where $f = 1$, the coefficient

$$a(\theta, x) = 5 + \sum_{1 \le i+j \le 4} \frac{1}{\sqrt{i^2 + j^2}} \theta_{i,j} \cos(i\pi x_1)\cos(j\pi x_2).$$



# Numerical results: Comparison

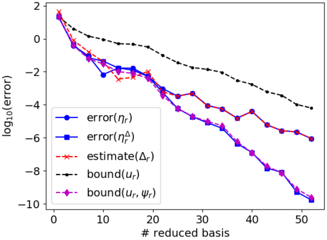



(a) Samples at step $l = 0, 9, 99$

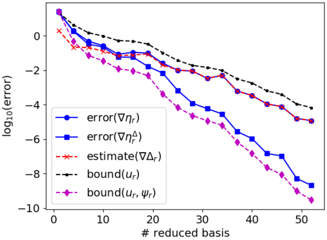
(b) marginal posterior

Figure: Comparison of (128) sample distribution driven by SVGD high-fidelity approximation (blue) and reduced basis approximation (red).

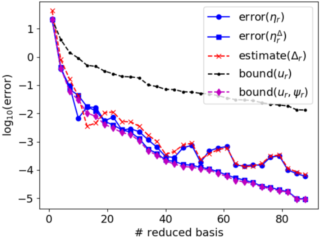
# Numerical results: Accuracy



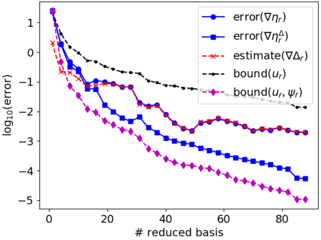

(a) adaptive construction $\eta_y$




(b) adaptive construction $\nabla_\theta \eta_y$




(c) fixed construction $\eta_y$




(d) fixed construction $\nabla_\theta \eta_y$

# Numerical results: Adaptive greedy algorithm

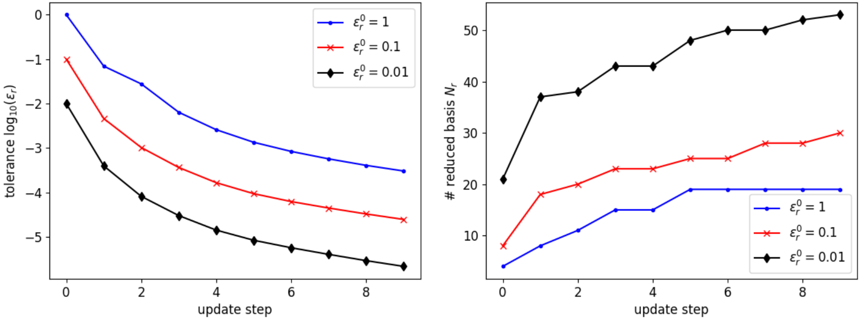



Figure: Tolerances for adaptive greedy algorithm (left);
# reduced basis functions for difference initial tolerances (right)

# Numerical results: Cost

| | | FE | adaptive RB | | | fixed RB |
|---|---|---|---|---|---|---|
| initial tolerance $\varepsilon_r^0$ | | n/a | $1$ | $0.1$ | $0.01$ | $0.00001$ |
| $M = 64$ | DOF $(N_h, N_r)$ | 16641 | 20 | 31 | 49 | 62 |
| | time to build RB | n/a | $4.4$ | $7.1$ | $12.2$ | $15.8$ |
| | time for evaluation | $1.8 \times 10^3$ | $4.4$ | $4.8$ | $5.8$ | $7.3$ |
| | speedup factor | 1 | 203 | 148 | 98 | 62 |
| M=128 | DOF $(N_h, N_r)$ | 16641 | 19 | 30 | 53 | 87 |
| | time to build RB | n/a | $4.5$ | $7.3$ | $14.3$ | $26.3$ |
| | time for evaluation | $3.5 \times 10^3$ | $8.3$ | $9.5$ | $11.8$ | $19.2$ |
| | speedup factor | 1 | 267 | 212 | 137 | 78 |

Table: Comparison of high fidelity and reduced basis approximations on degrees of freedom (DOF), CPU time for different tolerances and # samples

P. Chen, O. Ghattas. Stein variational reduced basis Bayesian inversion, 2019.

# Summary

**Take away message:**

- Reduced basis methods reduce the computational cost while preserving physical structure with certified accuracy.
- Leverage goal-oriented adaptive construction of RB.

**Ongoing:**

- RB for SVN.
- Parameter and state reduction by projected SV + RB.
- Extension to nonlinear and nonaffine problems.

Thank you for your attention!

# References I


Beskos, A., Girolami, M., Lan, S., Farrell, P. E., and Stuart, A. M. (2017).
Geometric mcmc for infinite-dimensional inverse problems.
*Journal of Computational Physics*, 335:327 – 351.

Binev, P., Cohen, A., Dahmen, W., DeVore, R., Petrova, G., and Wojtaszczyk, P. (2011).
Convergence rates for greedy algorithms in reduced basis methods.
*SIAM Journal on Mathematical Analysis*, 43(3):1457–1472.

Bui-Thanh, T., Ghattas, O., Martin, J., and Stadler, G. (2013).
A computational framework for infinite-dimensional bayesian inverse problems part I: The linearized case, with application to global seismic inversion.
*SIAM Journal on Scientific Computing*, 35(6):A2494–A2523.

# References II


Bui-Thanh, T. and Girolami, M. (2014).
Solving large-scale pde-constrained bayesian inverse problems with riemann manifold hamiltonian monte carlo.
*Inverse Problems*, 30(11):114014.

Chen, P. and Ghattas, O. (2019).
Stein variational reduced basis Bayesian inversion.
*in preparation*.

Chen, P. and Schwab, C. (2016).
Sparse-grid, reduced-basis Bayesian inversion: Nonaffine-parametric nonlinear equations.
*Journal of Computational Physics*, 316:470 – 503.

# References III


Chen, P., Villa, U., and Ghattas, O. (2017).
Hessian-based adaptive sparse quadrature for infinite-dimensional Bayesian inverse problems.
*Computer Methods in Applied Mechanics and Engineering*, 327:147–172.

Chen, P., Wu, K., Chen, J., O'Leary-Roseberry, T., and Ghattas, O. (2019).
Projected stein variational Newton: A fast and scalable Bayesian inference method in high dimensions.
*arXiv preprint arXiv:1901.08659*.

Cui, T., Marzouk, Y., and Willcox, K. (2015).
Data-driven model reduction for the bayesian solution of inverse problems.
*International Journal for Numerical Methods in Engineering*, 102(5):966–990.

# References IV


Detommaso, G., Cui, T., Marzouk, Y., Spantini, A., and Scheichl, R. (2018).
A stein variational Newton method.
In *Advances in Neural Information Processing Systems*, pages 9187–9197.

DeVore, R., Petrova, G., and Wojtaszczyk, P. (2012).
Greedy algorithms for reduced bases in Banach spaces.
*Arxiv preprint arXiv:1204.2290*.

El Moselhy, T. and Marzouk, Y. (2012).
Bayesian inference with optimal maps.
*Journal of Computational Physics*.

# References V


Farcas, I.-G., Latz, J., Ullmann, E., Neckel, T., and Bunagrtz, H.-J. (2019).
Multilevel adaptive sparse Leja approximations for Bayesian inverse problems.
*arXiv preprint arXiv:1904.12204*.

Gantner, R. N. and Schwab, C. (2016).
Computational higher order quasi-Monte Carlo integration.
In *Monte Carlo and Quasi-Monte Carlo Methods*, pages 271–288. Springer.

Girolami, M. and Calderhead, B. (2011).
Riemann manifold langevin and hamiltonian monte carlo methods.
*Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(2):123–214.

# References VI


Lan, S., Bui-Thanh, T., Christie, M., and Girolami, M. (2016).
Emulation of higher-order tensors in manifold monte carlo methods for bayesian inverse problems.
*Journal of Computational Physics*, 308:81–101.

Lassila, T., Manzoni, A., Quarteroni, A., and Rozza, G. (2013).
A reduced computational and geometrical framework for inverse problems in hemodynamics.
*International journal for numerical methods in biomedical engineering*, 29(7):741–776.

Lieberman, C., Willcox, K., and Ghattas, O. (2010).
Parameter and state model reduction for large-scale statistical inverse problems.
*SIAM Journal on Scientific Computing*, 32(5):2523–2542.

# References VII


Liu, Q. and Wang, D. (2016).
Stein variational gradient descent: A general purpose Bayesian inference algorithm.
In *Advances In Neural Information Processing Systems*, pages 2378–2386.

Martin, J., Wilcox, L., Burstedde, C., and Ghattas, O. (2012).
A stochastic Newton MCMC method for large-scale statistical inverse problems with application to seismic inversion.
*SIAM Journal on Scientific Computing*, 34(3):A1460–A1487.

Marzouk, Y., Najm, H., and Rahn, L. (2007).
Stochastic spectral methods for efficient bayesian solution of inverse problems.
*Journal of Computational Physics*, 224(2):560–586.

# References VIII


Marzouk, Y. and Xiu, D. (2009).
A stochastic collocation approach to Bayesian inference in inverse problems.
*Communications in Computational Physics*, 6(4):826–847.

Nguyen, C., Rozza, G., Huynh, D., and Patera, A. (2010).
Reduced basis approximation and a posteriori error estimation for parametrized parabolic pdes; application to real-time bayesian parameter estimation.
Technical report, John Wiley & Sons.

Oliver, D. S. (2017).
Metropolized randomized maximum likelihood for improved sampling from multimodal distributions.
*SIAM/ASA Journal on Uncertainty Quantification*, 5(1):259–277.

# References IX


Rezende, D. J. and Mohamed, S. (2015).
Variational inference with normalizing flows.
*arXiv preprint arXiv:1505.05770*.

Schillings, C. and Schwab, C. (2013).
Sparse, adaptive smolyak quadratures for bayesian inverse problems.
*Inverse Problems*, 29(6).

Schillings, C., Sprungk, B., and Wacker, P. (2019).
On the convergence of the Laplace approximation and noise-level-robustness of Laplace-based Monte Carlo methods for Bayesian inverse problems.
*arXiv preprint arXiv:1901.03958*.

Schwab, C. and Stuart, A. (2012).
Sparse deterministic approximation of bayesian inverse problems.
*Inverse Problems*, 28(4):045003.

Spantini, A., Bigoni, D., and Marzouk, Y. (2018).
Inference via low-dimensional couplings.
*The Journal of Machine Learning Research*, 19(1):2639–2709.

Stuart, A., Voss, J., and Wilberg, P. (2004).
Conditional path sampling of sdes and the langevin mcmc method.
*Communications in Mathematical Sciences*, 2(4):685–697.

Wang, J. and Zabaras, N. (2005).
Using Bayesian statistics in the estimation of heat source in radiation.
*International journal of heat and mass transfer*, 48(1):15–29.

# References XI


Wang, K., Bui-Thanh, T., and Ghattas, O. (2018).
A randomized maximum a posteriori method for posterior sampling of high dimensional nonlinear Bayesian inverse problems.
*SIAM Journal on Scientific Computing*, 40(1):A142–A171.

Wang, Z., Cui, T., Bardsley, J., and Marzouk, Y. (2019).
Scalable optimization-based sampling on function space.
*arXiv preprint arXiv:1903.00870*.


Thank you for your attention!