Scalable Bayesian Inference for Inverse Problems

Cheng Zhang

Joint work with Babak Shahbaba and Hongkai Zhao

February 25, 2019

Fred Hutchinson Cancer Research Center

Introduction

Inverse Problems

Given $y \in Y$, find $\theta \in X$, s.t.

$$y = \mathcal{G}(\theta)$$

- $\mathcal{G}$: observation operator (**forward model**)
- y: observed data

Inverse Problems

Given $y \in Y$, find $\theta \in X$, s.t.

$$y = \mathcal{G}(\theta)$$

- $\mathcal{G}$: observation operator (**forward model**)
- y: observed data

Examples

Inverse Problems

Given $y \in Y$, find $\theta \in X$, s.t.

$$y = \mathcal{G}(\theta)$$

- $\mathcal{G}$: observation operator (**forward model**)
- y: observed data

Examples

- Linear regression: $\mathcal{G}(\theta) = A\theta$

Inverse Problems

Given $y \in Y$, find $\theta \in X$, s.t.

$$y = \mathcal{G}(\theta)$$

- $\mathcal{G}$: observation operator (**forward model**)
- y: observed data

Examples

- Linear regression: $\mathcal{G}(\theta) = A\theta$
- Elliptic Inverse Problem

$$-\nabla \cdot (e^{\theta} \nabla u) = f, \quad x \in D$$
$$u = \phi, \quad x \in \partial D$$

$\mathcal{G}(\theta) = l(u_\theta)$, where l is some linear functional of u_θ.

Classical Approach

Inverse problems are typically *ill-posed*: no solution, solution not unique, sensitive on y.

Classical Approach

Inverse problems are typically *ill-posed*: no solution, solution not unique, sensitive on y.

Least-square

$$\arg\min_{\theta \in X} \frac{1}{2} \|y - \mathcal{G}(\theta)\|_Y^2$$

Classical Approach

Inverse problems are typically *ill-posed*: no solution, solution not unique, sensitive on y.

Least-square + **regularization**

$$\arg\min_{\theta \in X} \frac{1}{2}\|y - \mathcal{G}(\theta)\|_Y^2 + \frac{1}{2}\|\theta - \theta_0\|^2$$

Classical Approach

Inverse problems are typically *ill-posed*: no solution, solution not unique, sensitive on y.

Least-square + regularization

$$\arg\min_{\theta \in X} \frac{1}{2}\|y - \mathcal{G}(\theta)\|_Y^2 + \frac{1}{2}\|\theta - \theta_0\|^2$$

However, choice of norms and regularization are somewhat arbitrary.

The Bayesian Approach to Inverse Problems

A more appropriate model with noisy observations

$$y = \mathcal{G}(\theta) + \eta$$

- η: observational noise

The Bayesian Approach to Inverse Problems

A more appropriate model with noisy observations

$$y = \mathcal{G}(\theta) + \eta$$

- η: observational noise

Suppose $\eta \sim \rho(\eta)$, the data likelihood is

$$p(y|\theta) = \rho(y - \mathcal{G}(\theta))$$

The Bayesian Approach to Inverse Problems

A more appropriate model with noisy observations

$$y = \mathcal{G}(\theta) + \eta$$

- η: observational noise

Suppose $\eta \sim \rho(\eta)$, the data likelihood is

$$p(y|\theta) = \rho(y - \mathcal{G}(\theta))$$

Given prior $\theta \sim p(\theta)$, the **posterior** is

$$p(\theta|y) \propto p(y|\theta)p(\theta) = \rho(y - \mathcal{G}(\theta))p(\theta)$$

The Bayesian Approach to Inverse Problems

A more appropriate model with noisy observations

$$y = \mathcal{G}(\theta) + \eta$$

- η: observational noise

Suppose $\eta \sim \rho(\eta)$, the data likelihood is

$$p(y|\theta) = \rho(y - \mathcal{G}(\theta))$$

Given prior $\theta \sim p(\theta)$, the **posterior** is

$$p(\theta|y) \propto p(y|\theta)p(\theta) = \rho(y - \mathcal{G}(\theta))p(\theta)$$

Remark: classical approach (with regularization) can be viewed as *maximum a posterior estimate* (MAP).

Markov Chain Monte Carlo

Metropolis-Hastings

- draw a sample $\theta' \sim q(\theta'|\theta)$
- accept with probability $\alpha(\theta'|\theta) = \min\left(1, \frac{p(\theta'|y)q(\theta|\theta')}{p(\theta|y)q(\theta'|\theta)}\right)$.

Simple MCMCs are not efficient when parameters are correlated.

Markov Chain Monte Carlo

Metropolis-Hastings

- draw a sample $\theta' \sim q(\theta'|\theta)$
- accept with probability $\alpha(\theta'|\theta) = \min\left(1, \frac{p(\theta'|y)q(\theta|\theta')}{p(\theta|y)q(\theta'|\theta)}\right)$.

Simple MCMCs are not efficient when parameters are correlated.

Hamiltonian Monte Carlo

$$H(\theta, r) = -\log p(\theta|y) + \frac{1}{2} r^T M^{-1} r \Leftrightarrow p(\theta, r) = p(\theta|y) \cdot \mathcal{N}(r|0, M)$$

- draw an auxiliary momentum $r \sim \mathcal{N}(0, M)$
- simulate the Hamiltonian dynamics: $(\theta, r) \to (\theta', r')$

$$\frac{d\theta}{dt} = \nabla_r H, \quad \frac{dr}{dt} = -\nabla_\theta H$$

- accept with probability
  $\alpha(\theta', r'|\theta, r) = \min(1, \exp[H(\theta, r) - H(\theta', r')])$

Scalable Bayesian inference via Surrogate Methods

Computational Challenges

Potential energy function

$$\begin{aligned} U(\theta) &\triangleq -\log p(\theta|y) \\ &= -\log \rho(y - \mathcal{G}(\theta)) - \log p(\theta) \end{aligned}$$

Challenges

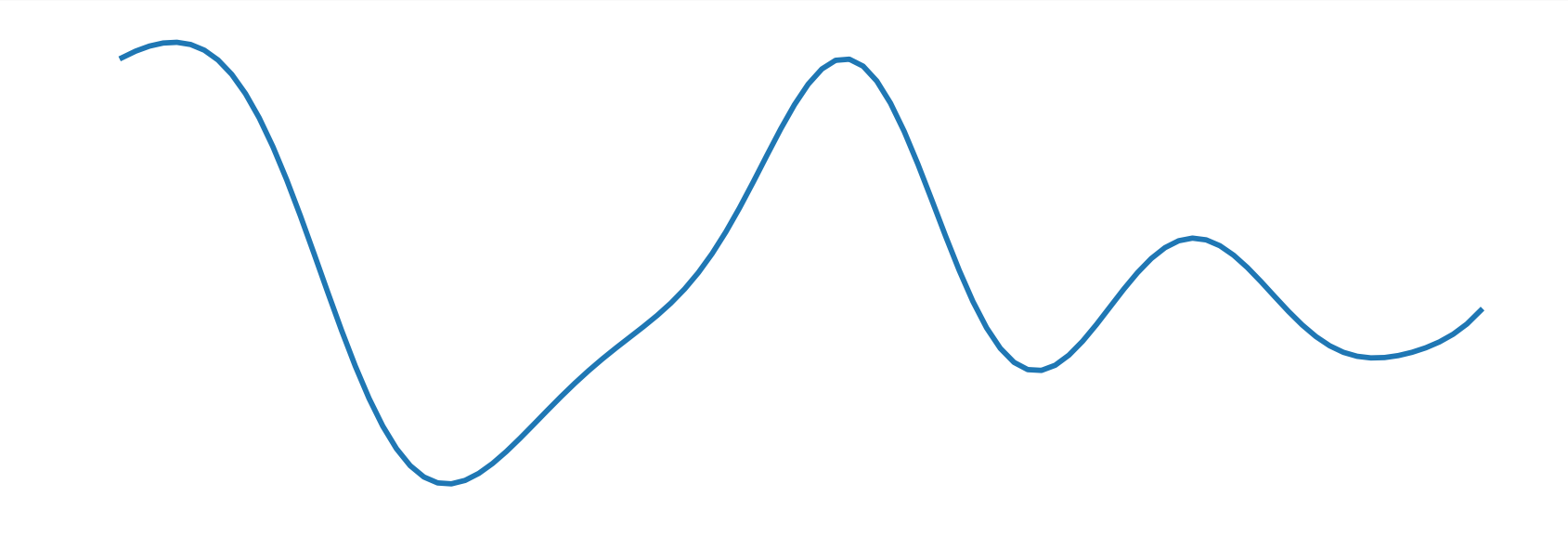
- dependency on θ is unknown, complicated posterior
- forward model $\mathcal{G}(\theta)$ is computationally expensive, $U(\theta), \nabla_\theta U(\theta)$ are hard to evaluate

Surrogate Methods

Idea: exploit the regularity of the probabilistic model

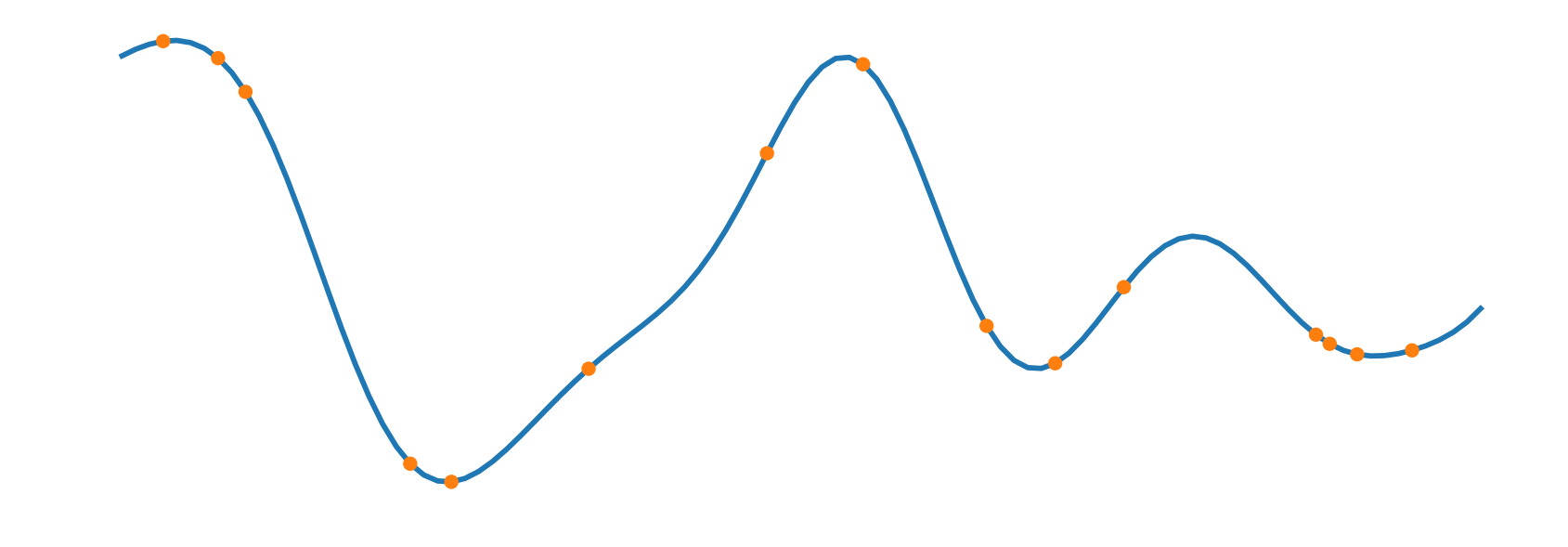
Surrogate Methods

Idea: exploit the regularity of the probabilistic model



Surrogate Methods

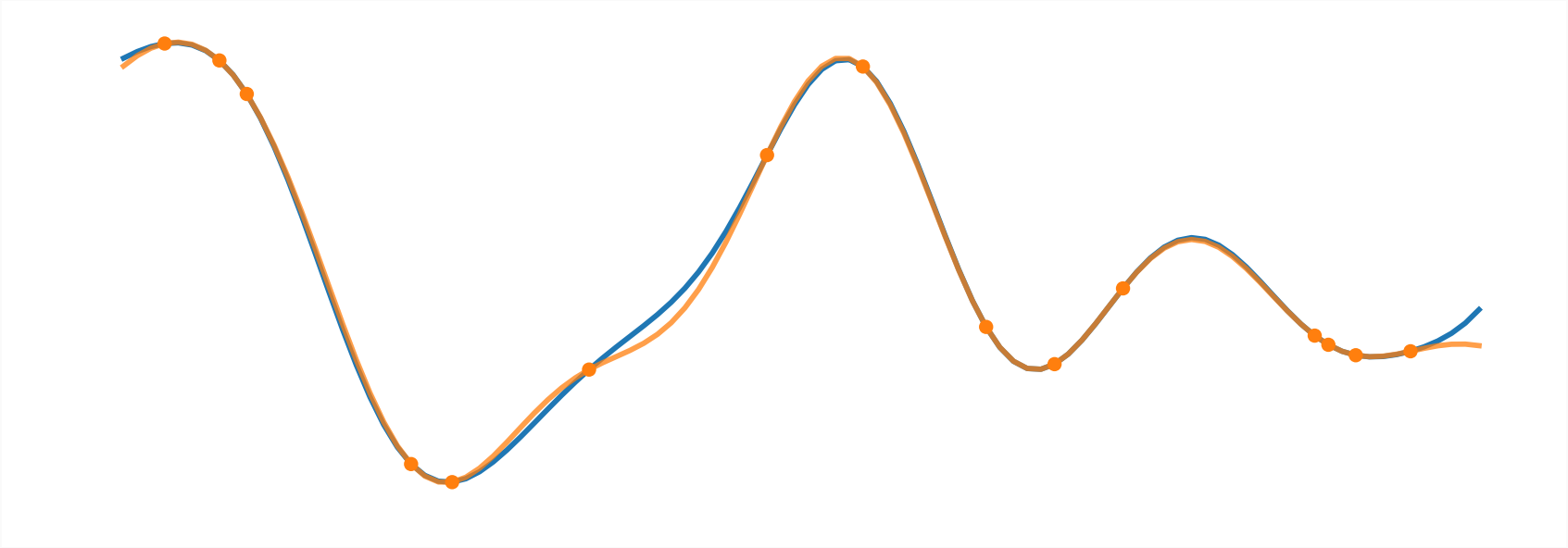
Idea: exploit the regularity of the probabilistic model



Surrogate Methods

Idea: exploit the regularity of the probabilistic model

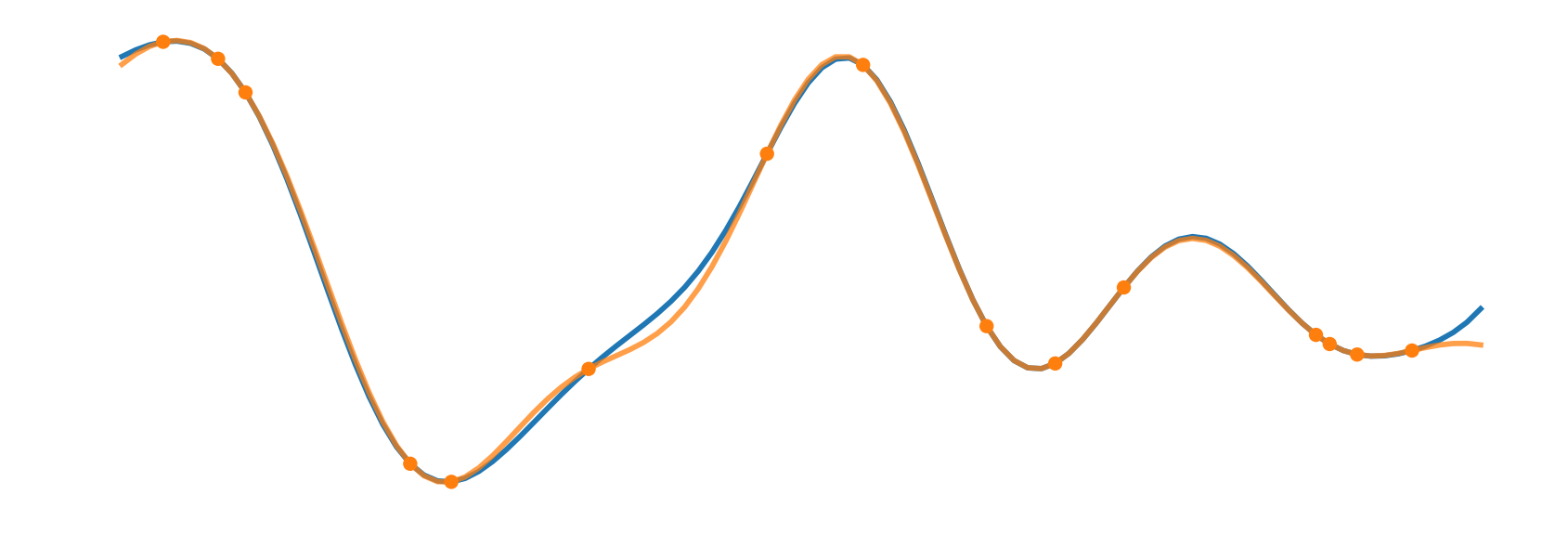
$$U(\theta) \approx U^S(\theta), \quad \nabla_\theta U(\theta) \approx \nabla_\theta U^S(\theta)$$



Surrogate Methods

Idea: exploit the regularity of the probabilistic model

$$U(\theta) \approx U^S(\theta), \quad \nabla_\theta U(\theta) \approx \nabla_\theta U^S(\theta)$$



First suggested by Neal, Liu in 90s. Examples: Gaussian Processes (Rasmussen 2003, Lan et al 2016), Reproducing Kernel Hilbert Spaces (Strathmann et al 2015), Random Networks/Bases (Zhang et al 2017).

Random Bases Surrogate

$$U_\psi^S(\theta) = \sum_{i=1}^{S} \psi_i a(\theta; \gamma_i), \quad \gamma_i \sim q(\gamma)$$

Train by matching the function values or gradient values

potential matching

$$\hat{\psi} = \arg\min_{\psi, b} \sum_{j=1}^{M} \|U_\psi^S(\theta_j) - U(\theta_j) - b\|^2$$

score matching

$$\hat{\psi} = \arg\min_{\psi} \sum_{j=1}^{M} \|\nabla_\theta U_\psi^S(\theta_j) - \nabla_\theta U(\theta_j)\|^2$$

where $T = \{\theta_1, \ldots, \theta_M\}$ is the training set (e.g., data from burn-in).

Random Bases Surrogate

$$U_\psi^S(\theta) = \sum_{i=1}^{S} \psi_i a(\theta; \gamma_i), \quad \gamma_i \sim q(\gamma)$$

Train by matching the function values or gradient values

potential matching

$$\hat{\psi} = \arg\min_{\psi, b} \sum_{j=1}^{M} \|U_\psi^S(\theta_j) - U(\theta_j) - b\|^2$$

score matching

$$\hat{\psi} = \arg\min_{\psi} \sum_{j=1}^{M} \|\nabla_\theta U_\psi^S(\theta_j) - \nabla_\theta U(\theta_j)\|^2$$

where $T = \{\theta_1, \ldots, \theta_M\}$ is the training set (e.g., data from burn-in).

Why use random bases?

Random Bases Surrogate

$$U_\psi^S(\theta) = \sum_{i=1}^{S} \psi_i a(\theta; \gamma_i), \quad \gamma_i \sim q(\gamma)$$

Train by matching the function values or gradient values

potential matching

$$\hat{\psi} = \arg\min_{\psi, b} \sum_{j=1}^{M} \|U_\psi^S(\theta_j) - U(\theta_j) - b\|^2$$

score matching

$$\hat{\psi} = \arg\min_{\psi} \sum_{j=1}^{M} \|\nabla_\theta U_\psi^S(\theta_j) - \nabla_\theta U(\theta_j)\|^2$$

where $T = \{\theta_1, \ldots, \theta_M\}$ is the training set (e.g., data from burn-in).

Why use random bases?

- scales linearly with M, while *GPs* and *RKHS* scale cubically.

Random Bases Surrogate

$$U_\psi^S(\theta) = \sum_{i=1}^{S} \psi_i a(\theta; \gamma_i), \quad \gamma_i \sim q(\gamma)$$

Train by matching the function values or gradient values

potential matching

$$\hat{\psi} = \arg\min_{\psi, b} \sum_{j=1}^{M} \|U_\psi^S(\theta_j) - U(\theta_j) - b\|^2$$

score matching

$$\hat{\psi} = \arg\min_{\psi} \sum_{j=1}^{M} \|\nabla_\theta U_\psi^S(\theta_j) - \nabla_\theta U(\theta_j)\|^2$$

where $T = \{\theta_1, \ldots, \theta_M\}$ is the training set (e.g., data from burn-in).

Why use random bases?

- scales linearly with M, while *GPs* and *RKHS* scale cubically.
- theoretical guarantee for good approximation (Rahimi and Recht 2008): $\forall f$, with probability $1 - \delta$, $\exists \psi$ s.t.

$$\|U_\psi^S - f\| \leq \frac{\|f\|}{\sqrt{S}} \left(1 + \sqrt{2 \log \frac{1}{\delta}}\right)$$

Surrogate Induced Hamiltonian Flow

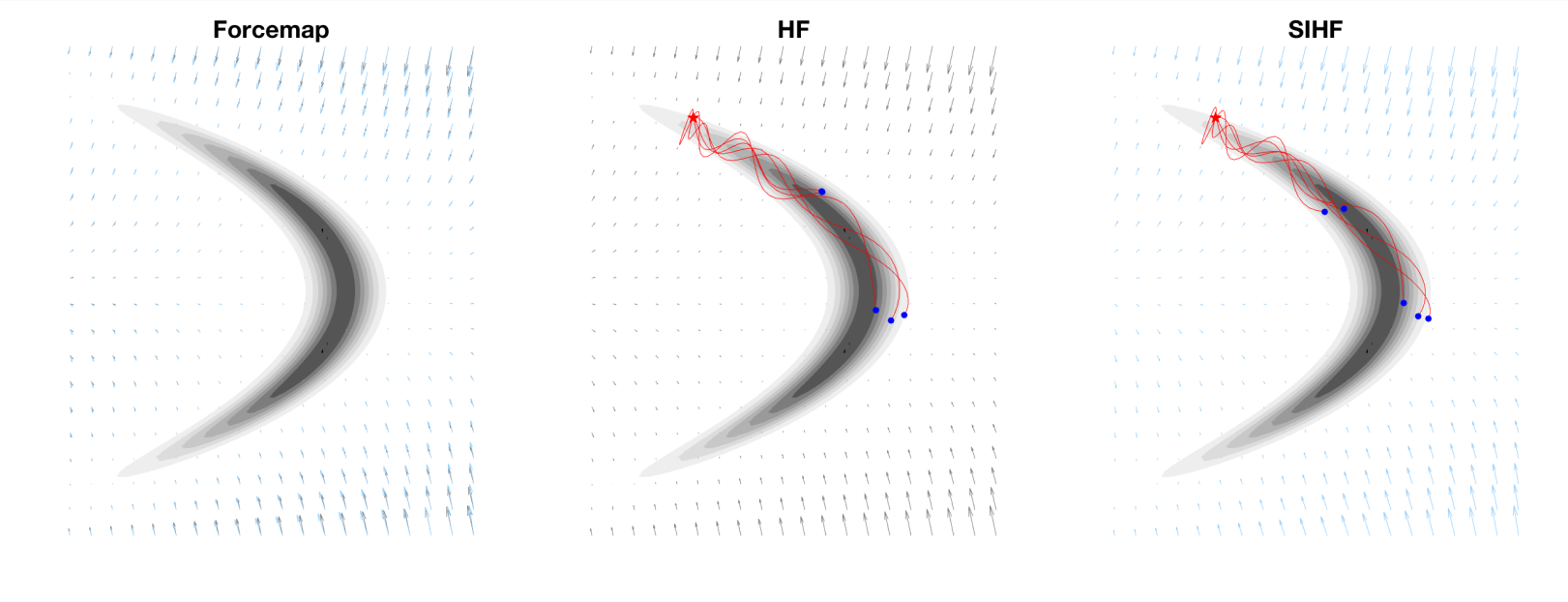
Define $H^S_\psi(\theta, r) = U^S_\psi(\theta) + \frac{1}{2} r^T M^{-1} r$, surrogate induced Hamilton's equations

$$\frac{d\theta}{dt} = M^{-1} r, \quad \frac{dr}{dt} = -\nabla_\theta U^S_\psi(\theta)$$

Surrogate Induced Hamiltonian Flow

Define $H^S_\psi(\theta, r) = U^S_\psi(\theta) + \frac{1}{2} r^T M^{-1} r$, surrogate induced Hamilton's equations

$$\frac{d\theta}{dt} = M^{-1} r, \quad \frac{dr}{dt} = -\nabla_\theta U^S_\psi(\theta)$$



An Elliptic PDE Inverse Problem

Let κ be the diffusion function and u be the pressure field

$$-\nabla \cdot (\kappa \nabla u) = 0, \quad x \in [0,1]^2$$
$$u(x_1, 0) = x_1,\ u(x_1, 1) = 1 - x_1,\ \partial_{x_1} u(0, x_2) = \partial_{x_1} u(1, x_2) = 0$$

A log-Gaussian prior is used for κ

$$K(x, y) = \sigma^2 \exp\left(-\frac{\|x - y\|_2^2}{2\ell^2}\right)$$

parameterize diffusivity field with Karhunen-Loeve (K-L) expansion

$$\kappa_\theta(x) \approx \exp\left(\sum_{i=1}^{d} \theta_i \sqrt{\lambda_i} v_i(x)\right)$$

noisy observations

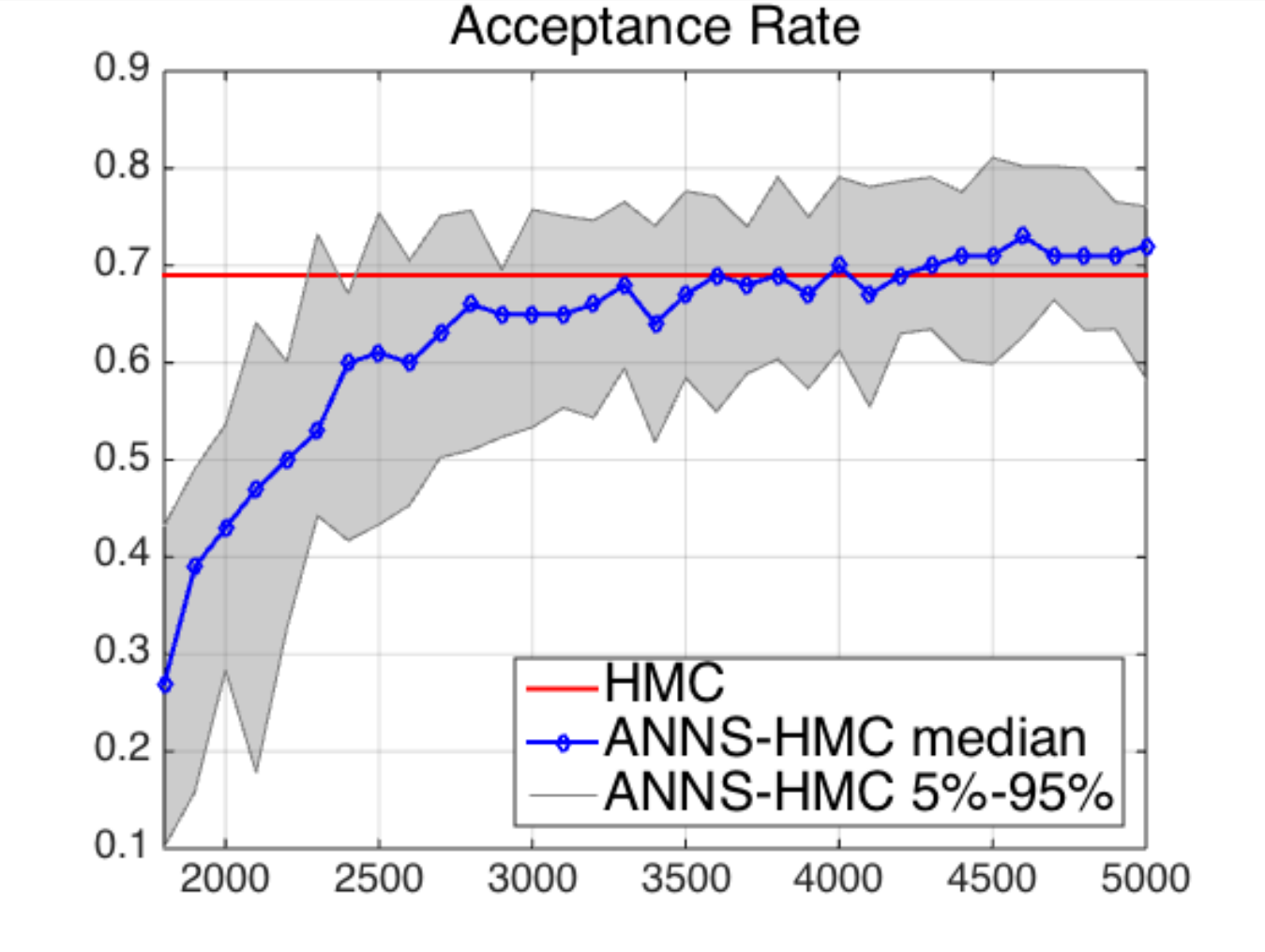
$$y_j = u_\theta(x_j) + \eta_j, \quad j = 1, \ldots, J$$

Results

Table 1: Comparing HMC, Riemannian Manifold HMC (RMHMC) and random bases surrogate accelerations. For each method, we provide the acceptance probability (AP), the effective sample size (ESS), the CPU time (s) for each iteration and the time-normalized ESS.

Method	AP	ESS	s/Iter	$\min(\mathrm{ESS})/\mathrm{s}$	spdup
HMC	0.91	(4533,5000,5000)	0.775	1.17	1
RMHMC	0.80	(5000,5000,5000)	4.388	0.23	0.20
RNS-HMC	0.75	(2306,3034,3516)	0.066	7.10	**6.07**
RNS-RMHMC	0.66	(2126,4052,5000)	0.097	4.38	3.74

Adaptive Training



Conclusion

- We proposed **random bases surrogate methods**, an efficient scalable Bayesian approach for inverse problems.
- Random bases surrogates properly exploit regularity of probabilistic models, and remain data efficient. More efficiency can be obtained when used adaptively.
- Surrogate methods can be used for big data problems as well. Moreover, surrogate methods lead to a natural combination of variational inference and MCMC (Zhang et al 2018).
- We can incorporate more flexible approximating architectures in surrogate construction.

Questions?

Bias, Variance, and Computation Trade-off