# Adaptive Model Reduction for Large-Scale Bayesian Inverse Problems

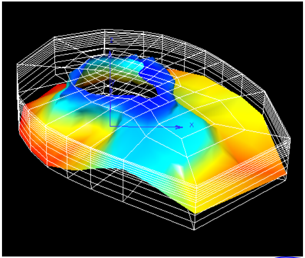
Tiangang Cui[1], Youssef Marzouk[2], Karen Willcox[3]

[1]Monash University
[2]Massachusetts Institute of Technology
[3]University of Texas at Austin

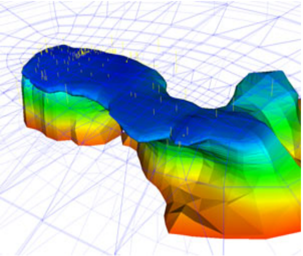
Chengdu, China, September, 2018

# Inverse Problems

Parameter $\mathbb{X}$



State $\mathbb{U}$



Data $\mathbb{Y}$

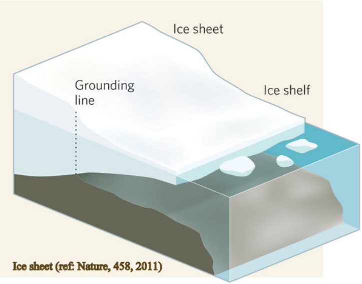
$$\text{s.t.} \quad A(u, x) = 0 \qquad\qquad y_o = C(u, e)$$

- From left to right: Forward Model $F : \mathbb{X} \to \mathbb{Y}$
- From right to left: Inverse problem
- State $u$ is high-dimensional for numerical accuracy
- Parameter $x$ can be high-dimensional for resolving spatial heterogeneity
- Data are indirect and noisy, often incomplete for estimating $x$
- Ill-posedness $\Longrightarrow$ non-uniqueness and uncertainty

# Example: Arolla Glacier

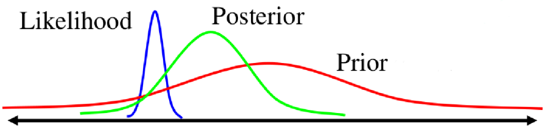
Goal: estimating basal sliding coefficients from surface velocity measurements.

$$
\begin{aligned}
-\boldsymbol{\nabla}\cdot[2\eta(\boldsymbol{u})\,\dot{\boldsymbol{\varepsilon}}_{\boldsymbol{u}}-\boldsymbol{I}p]&=\rho\boldsymbol{g} && \text{in } \Omega\\
\boldsymbol{\nabla}\cdot\boldsymbol{u}&=0 && \text{in } \Omega\\
\boldsymbol{\sigma}_{\boldsymbol{u}}\boldsymbol{n}&=\boldsymbol{0} && \text{on } \Gamma_t\\
\boldsymbol{u}\cdot\boldsymbol{n}&=0 && \text{on } \Gamma_b\\
\boldsymbol{T}\boldsymbol{\sigma}_{\boldsymbol{u}}\boldsymbol{n}+\exp(x)\boldsymbol{T}\boldsymbol{u}&=\boldsymbol{0} && \text{on } \Gamma_b
\end{aligned}
$$




- $\boldsymbol{u}$ ice flow velocity, $p$ pressure
- $\boldsymbol{\sigma}_{\boldsymbol{u}}=-\boldsymbol{I}p+2\eta(\boldsymbol{u})\dot{\boldsymbol{\varepsilon}}_{\boldsymbol{u}}$ stress tensor
- $\dot{\boldsymbol{\varepsilon}}_{\boldsymbol{u}}=\frac{1}{2}(\boldsymbol{\nabla}\boldsymbol{u}+\boldsymbol{\nabla}\boldsymbol{u}^\top)$ strain rate tensor
- $\eta(\boldsymbol{u})=\frac{1}{2}A^{-\frac{1}{n}}\,\dot{\varepsilon}_{\mathrm{II}}^{\frac{1-n}{2n}}$ effective viscosity
- $\dot{\varepsilon}_{\mathrm{II}}=\frac{1}{2}\mathrm{tr}(\dot{\boldsymbol{\varepsilon}}_{\boldsymbol{u}}^2)$ second invariant of the strain rate tensor

- $\rho$ density, $\boldsymbol{g}$ gravity
- $\boldsymbol{n}$ unit normal vector
- $x$ log basal sliding coefficient
- $\boldsymbol{T}=\boldsymbol{I}-\boldsymbol{n}\otimes\boldsymbol{n}$ tangential operator
- $\Gamma_t$ and $\Gamma_b$ top and base boundaries




$$\textbf{Bayes' Rule} \qquad \underbrace{\pi\left(x|y_o\right)}_{Posterior} \propto \underbrace{L\left(y_o|F(x)\right)}_{Likelihood} \times \underbrace{\pi_0\left(x\right)}_{Prior}$$

- **Prior**: Expert knowledge or smooth assumptions based on spatial statistics: e.g. Gaussian Markov Random field and Gaussian process
- **Likelihood**: knowledge of the noise $e$, quantifies the probability of data $y_o$ being true for a given $x$. E.g., assuming $e$ follows Gaussian distribution, $e \sim \mathcal{N}(0, \Gamma_{\text{obs}})$

$$L\left(y_o|F(x)\right) \propto \exp\left(-\frac{1}{2}\left\|{\Gamma_{\text{obs}}}^{-\frac{1}{2}}\left[y_o - F(x)\right]\right\|^2\right)$$

- Posterior is an update from prior, using likelihood fucntion.

# Inverse Problems: Expectation

Summarize information over the posterior distribution by calculating the expected value of function of interest

$$\mathsf{E}_\pi\left[g\left(x\right)\right] = \int_{\mathbb{X}} g\left(x\right)\pi\left(x|y_o\right)\, dx$$

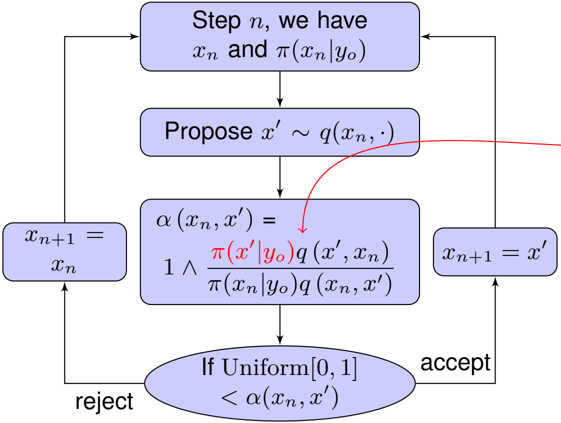
Example: mean $\mathsf{E}_\pi\left[x\right]$, variance $\mathsf{Var}_\pi\left[x\right]$ ...

- High-dimensional integrals $\Rightarrow$ Monte Carlo integration

$$x_1, \ldots, x_n \sim \pi(\cdot|y_o) \qquad\qquad \mathsf{E}_\pi\left[g(x)\right] \approx \frac{1}{n}\sum_{i=1}^{n} g(x_i)$$

- Use MCMC, SMC or importance sampling to get samples. We have to evaluate the posterior many times

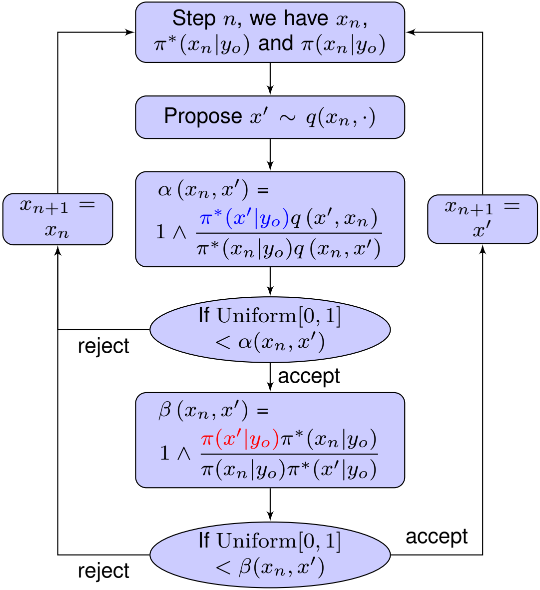
# MCMC Sampling




- Requires many iterations.
- Expensive model evaluation $A(x')$
- Each $x_n$ is a sample from the posterior $\Longrightarrow$ surrogates?
- Surrogate (ROM):

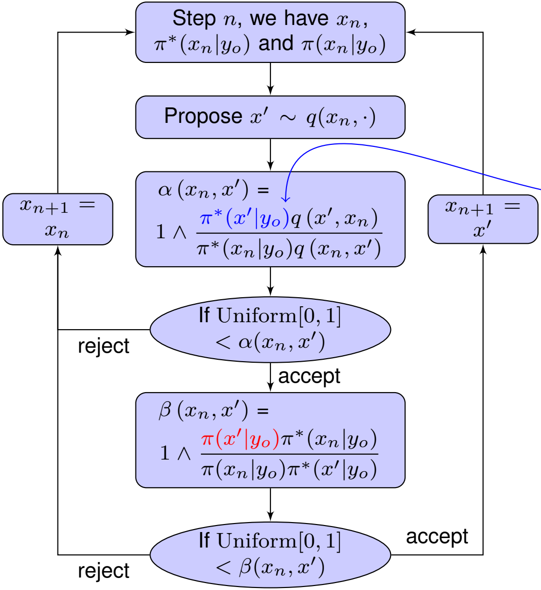
$$A^*(x) \approx A(x)$$

# Adaptive Delayed Acceptance



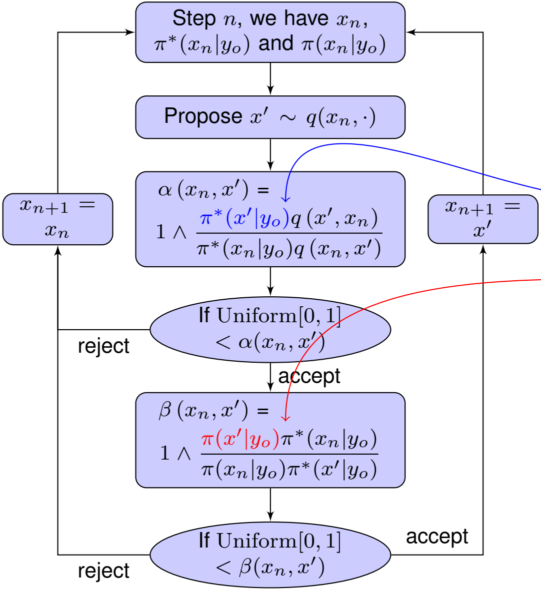

- Using a ROM $A^*(x)$, we have a fast $\pi^*(x|y_o) \approx \pi(x|y_o)$
- Fast acceptance/rejection $A^*(x')$
- Using the full model to ensure sampling the exact posterior
- Using new sample to update the reduced order model

# Adaptive Delayed Acceptance



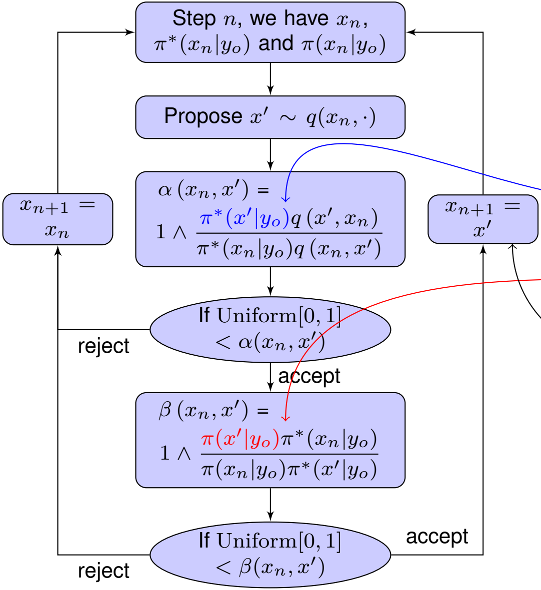

- Using a ROM $A^*(x)$, we have a fast $\pi^*(x|y_o) \approx \pi(x|y_o)$
- Fast acceptance/rejection
  $A^*(x')$
- Using the full model to ensure sampling the exact posterior
- Using new sample to update the reduced order model

# Adaptive Delayed Acceptance




- Using a ROM $A^*(x)$, we have a fast $\pi^*(x|y_o) \approx \pi(x|y_o)$
- Fast acceptance/rejection
  $A^*(x')$
- Using the full model to ensure
  sampling the exact posterior
- Using new sample to update
  the reduced order model

# Adaptive Delayed Acceptance




- Using a ROM $A^*(x)$, we have a fast $\pi^*(x|y_o) \approx \pi(x|y_o)$
- Fast acceptance/rejection $A^*(x')$
- Using the full model to ensure sampling the exact posterior
- Using new sample to update the reduced order model

Analyzed by Chen and Liu (1998), Christen and Fox (2005), and Cui et al. (2010)

# Model Reduction: Background

Consider the PDE model

$$\underbrace{B(x)u}_{Linear} + \underbrace{G(x,u)}_{Nonlinear} = 0.$$

$u \in \mathbb{R}^{N_s}$, $N_s$ is usually large.

## Reduced basis

For a target region of the parameter space, suppose the corresponding state $u(x)$ can be captured by an $r$-dimensional subspace, spanned by $\Phi \in \mathbb{R}^{N_s \times r}$, $r << N_s$.

## Reduced order model

Approximate solution $u(x) \approx \Phi u_r(x)$, a smaller system of equations:

$$\text{Galerkin}: \quad \Phi^\top \left[ B(x)\Phi u_r + G(x, \Phi u_r) \right] = 0,$$

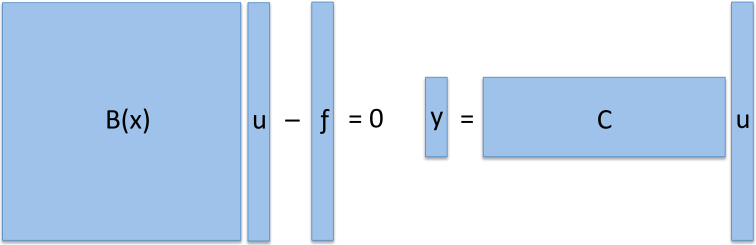
$G(x, \Phi u_r)$ can be handled by discrete empirical interpolation methods (DEIM)[a] or mission point method[b] ...

[a] Chaturantabut & Sorensen, SIAM Journal on Scientific Computing, 2010

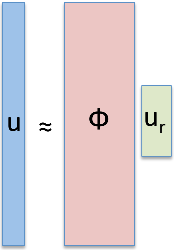
[b] Astrid et al., IEEE Transactions Automatic Control, 2008

# Model Reduction: Example

Poisson's Equation: $-\nabla \cdot (k(x)\nabla u) = f$ and observation operator $C \implies y = F(x)$




Given a reduced basis $\Phi$, approximate the state



Then apply Galerkin projection

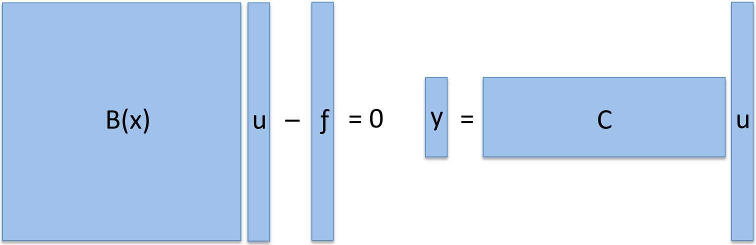
$$\Phi^T \left[ B(x) \, \Phi \, u_r - f \right] = 0$$

Reduced observation operator

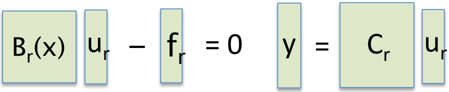
$$y = \left[ C \, \Phi \right] u_r$$

# Model Reduction: Example

Given $B_r(x) = \Phi^\top B(x)\Phi,\ f_r = \Phi^\top f,\ C_r = C\Phi$, the full model



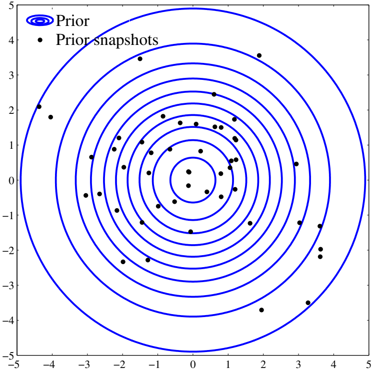

is reduced to




Reduced forward model $y_r = F^*(x)$.

The key is to identify the reduced basis $\Phi$.

- Generate parameter samples $x_i, \ldots, x_m$, solve $A(u_i, x_i) = 0$ to obtain snapshots of states $\{u_1, \ldots, u_m\}$.
- Orthogonalize the snapshots to get basis $\Phi$.
- Traditionally, snapshots are computed at prior samples*.
- However, the support of the posterior can be dramatically different from the prior.
- We designed a new model reduction approach to adaptively select snapshots from posterior.




* Wang & Zabaras, Int. J. Heat Mass Transfer, 2004

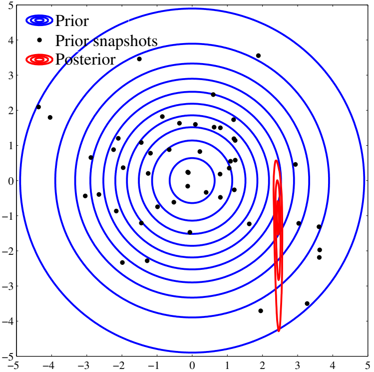
* Lieberman, Willcox, & Ghattas, SIAM Journal on Scientific Computing, 2010

* Lipponen, Seppnen & Kaipio, Inverse Problems Imaging, 2013

# Adaptive Model Reduction

The key is to identify the reduced basis $\Phi$.

- Generate parameter samples $x_i, \ldots, x_m$, solve $A(u_i, x_i) = 0$ to obtain snapshots of states $\{u_1, \ldots, u_m\}$.
- Orthogonalize the snapshots to get basis $\Phi$.
- Traditionally, snapshots are computed at prior samples*.
- However, the support of the posterior can be dramatically different from the prior.
- We designed a new model reduction approach to adaptively select snapshots from posterior.




* Wang & Zabaras, Int. J. Heat Mass Transfer, 2004

* Lieberman, Willcox, & Ghattas, SIAM Journal on Scientific Computing, 2010

* Lipponen, Seppnen & Kaipio, Inverse Problems Imaging, 2013

# Online Construction of ROM

Consider Poisson's Equation $-\nabla \cdot (k(x)\nabla u) = f$. Given partial observation of $u$, wish to reconstruct the diffusivity $k$, parametrized by $x$.

**Full model**

$$B(x)u(x) = f, \quad y(x) = Cu(x),$$

$C$: observation operator, $d$: model outputs.

**Reduced order model (ROM)**

Given reduced basis $V$, we have

$$\underbrace{\Phi^\top B(x)\Phi}_{B_r(x)} u_r(x) = \underbrace{\Phi^\top f}_{f_r}, \quad y_r(x) = \underbrace{C\Phi}_{C_r} u_r(x).$$

# Error Indicator: Dual Weighted Residual

We want to estimate the true error

$$t(x) = Cu(x) - C\Phi u_r(x)$$

without solving the full model.

## Dual Weighted Residual

- Dual solution $\gamma(x) = B(x)^{-\top} C^{\top}$
- Residual $r(x) = f - B(x)\Phi u_r(x)$
- The true error is given by

$$\begin{aligned} \gamma(x)^{\top} r(x) &= CB(x)^{-1}[f - B(x)\Phi u_r(x)] \\ &= Cu(x) - C\Phi u_r(x) \\ &= t(x) \end{aligned}$$

The dual solution $\gamma$ provides a way to quantify the impact of residual on the true error.

# Error Indicator: Approximated Dual Weighted Residual

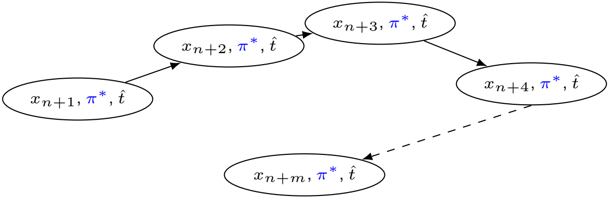
- Computing the exact dual solution $\gamma(x)$ for each $x$ is not feasible.
- Meyer and Matthies (2003) approximate the dual solution by using a ROM that has higher order of accuracy.
- In our setting, the maximum *a posteriori* estimate (MAP) provides a good estimate of the dual solution:

$$\hat{\gamma} \approx \gamma(x_{MAP})$$

- We can also use full model evaluations at posterior samples to build a library of dual solutions.

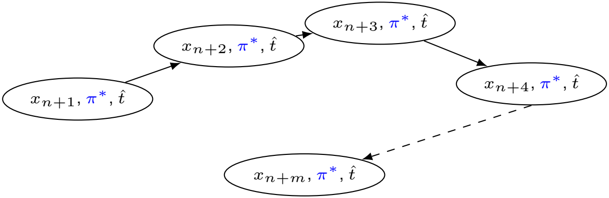
From $x_n$, sampling $\pi^*$ for m iterations

Estimate the error $\hat{t}(x_{n+i})$



From $x_n$, sampling $\pi^*$ for m iterations

Estimate the error $\hat{t}(x_{n+i})$




If $|\hat{t}| > \epsilon$ or $i > m$, evaluate $\pi$, and $\beta$

If $|\hat{t}| > \epsilon$, update ROM

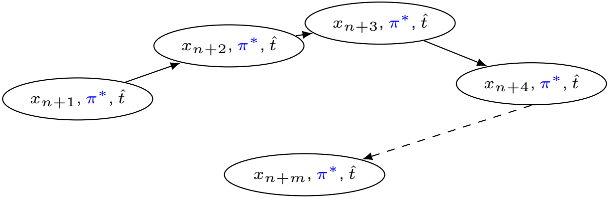
Iterate forward

- The Gram-Schmidt procedure is used to update the reduced basis vectors for a new snapshot.
- The above procedure samples the exact posterior, because of the correction using $\pi$, and $\beta$.

# Approximate Algorithm

From $x_n$, sampling $\pi^*$ for m iterations

Estimate the error $\hat{t}(x_{n+i})$



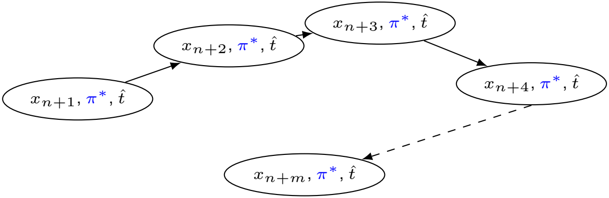

If $|\hat{t}| > \epsilon$ or $i > m$, evaluate $\pi$, and $\beta$

If $|\hat{t}| > \epsilon$, update ROM

Iterate forward

From $x_n$, sampling $\pi^*$ for m iterations

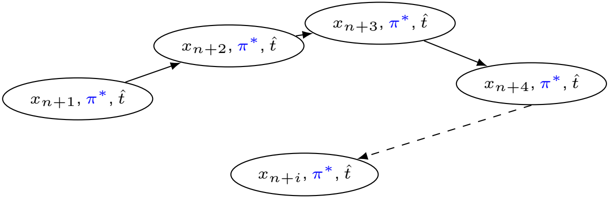
Estimate the error $\hat{t}(x_{n+i})$




If $|\hat{t}| > \epsilon$ or $i > m$, evaluate $\pi$, and $\beta$

If $|\hat{t}| > \epsilon$, update ROM

Iterate forward




If $|\hat{t}| > \epsilon$, evaluate $\pi$, and update ROM

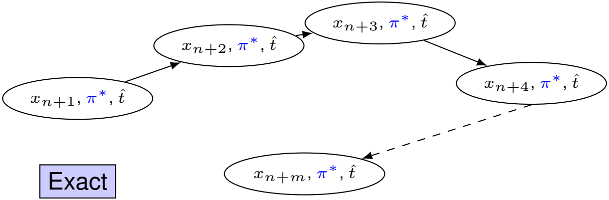
Otherwise, $\pi^* \approx \pi$

Iterate forward

# Approximate Algorithm

From $x_n$, sampling $\pi^*$ for m iterations

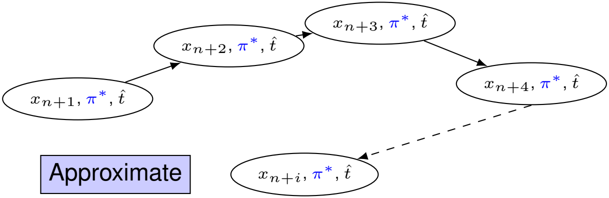
Estimate the error $\hat{t}(x_{n+i})$




If $|\hat{t}| > \epsilon$ or $i > m$, evaluate $\pi$, and $\beta$

If $|\hat{t}| > \epsilon$, update ROM

Iterate forward




If $|\hat{t}| > \epsilon$, evaluate $\pi$, and update ROM

Otherwise, $\pi^* \approx \pi$

Iterate forward

# Mean Square Error

The approximate algorithm **does not** sample from the exact posterior. However

## Mean Square Error

Given samples $x_i \sim \pi(\cdot|d)$, for some estimator

$$\hat{g} = \frac{1}{N}\sum_{i=1}^{N} g(x_i) \approx \int g(x)\pi(x|y_o)dx$$

The mean square error

$$MSE(\hat{g}) = Var(\hat{g}) + Bias(\hat{g})^2$$

- $Bias(\hat{\theta})^2 = 0$ for standard MCMC and the exact algorithm.
- $Bias(\hat{\theta})^2 \neq 0$ for the approximate algorithm. But

  $$Bias(\hat{\theta})^2 < C\epsilon^2$$

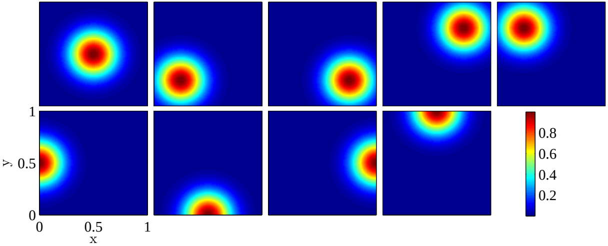
  Using Hellinger distance
- $Var(\hat{\theta}) = \dfrac{Var(\theta)}{ESS}$ dominates the MSE for small $\epsilon$, because the effective sample size (ESS) is usually small.

# Example 1: A 9D Test Case

In the domain $r \in [0,1]^2$, try to infer the diffusivity

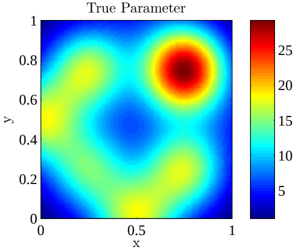
$$k(r) = \sum_{i=1}^{9} b_i(r) x_i$$

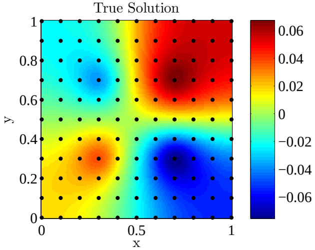
$$\log(x_i) \sim \mathcal{N}(\mu_i, \sigma_i^2)$$




121 potential measurements, signal to noise ratio 50.

Full model has $120 \times 120$ elements.



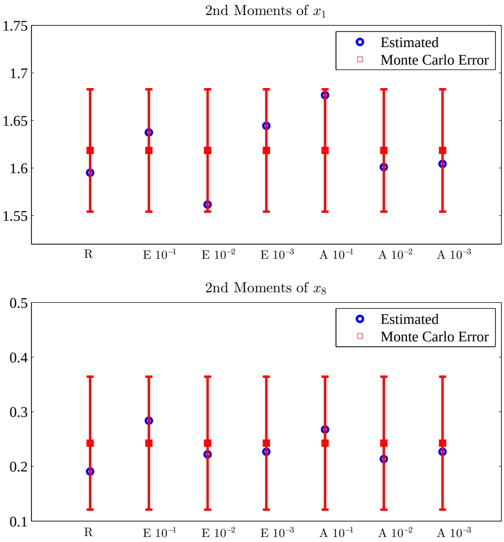
# Example 1: Sampling Efficiency

| | Reference | Exact | | | Approximate | | |
|---|---|---|---|---|---|---|---|
| Error threshold $\epsilon$ | - | $10^{-1}$ | $10^{-2}$ | $10^{-3}$ | $10^{-1}$ | $10^{-2}$ | $10^{-3}$ |
| Basis vectors | - | $14$ | $33$ | $57$ | $17$ | $35$ | $57$ |
| ESS / CPU time | $0.058$ | $2.5$ | $2.7$ | $2.6$ | 15 | $12$ | $8.9$ |
| Speed-up factor | $1$ | $43$ | $46$ | $45$ | $256$ | $213$ | $154$ |

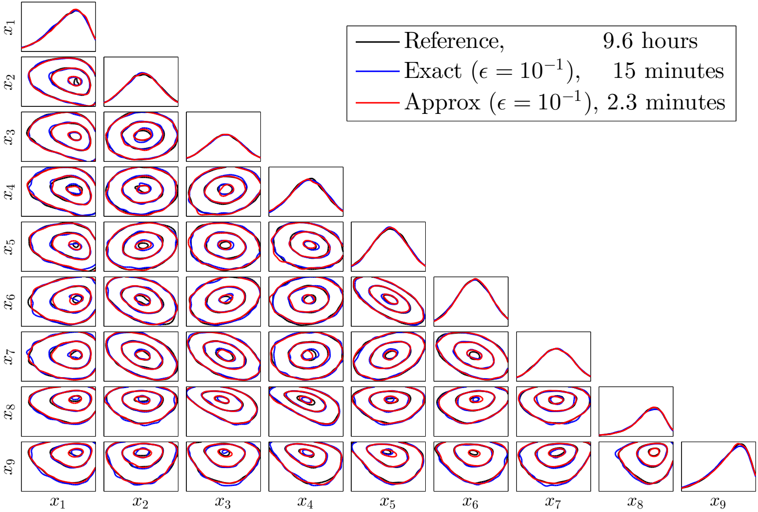
- Run both algorithms for $5 \times 10^5$ iterations, with $\epsilon = 10^{-1}, 10^{-2}, 10^{-3}$.
- $\epsilon$ is normalized by the standard derivation measurement noise.
- A reference MCMC (only based on the full model) is simulated for $5 \times 10^5$ iterations.
- Speed-up factor is estimated from CPU time per effective sample.

# Example 1: Sampling Accuracy

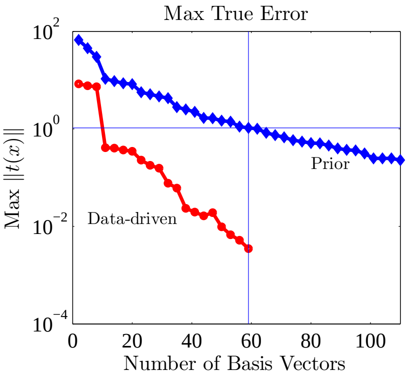
- Statistic of interest: variance of $x_1$ and $x_8$.
- Blue circle: estimator given by each chain.
- Error bar: $\pm 2$ s.t.d. of the Monte Carlo error of the estimator, 50 reference chains with $5 \times 10^5$ iterations.

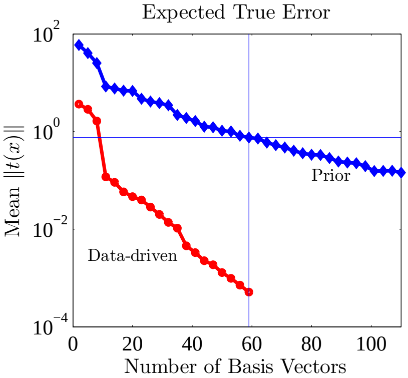


# Example 1: Sampling Accuracy



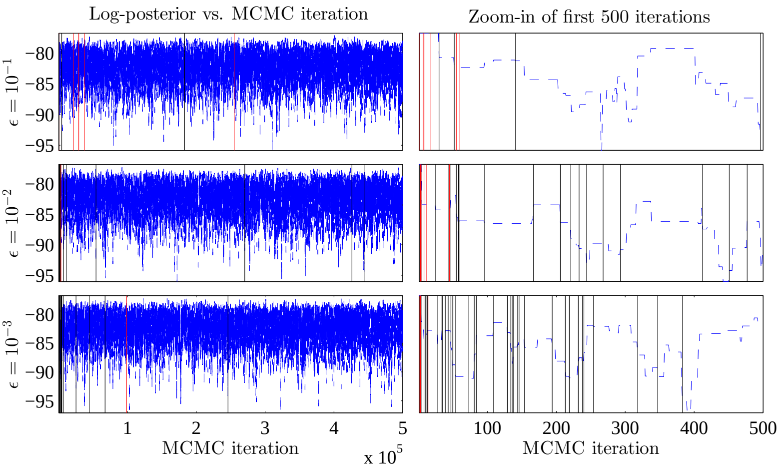
# Example 1: Accuracy of the ROM




- For benchmarking, $10^4$ snapshots from the prior to construct the ROM.
- The data-driven ROM are built with $\epsilon = 10^{-3}$.
- The true error for both ROMs are calculated on $10^4$ posterior samples.
- The true error is normalized by the standard derivation of measurement noise.

# Example 1: Numerical Results

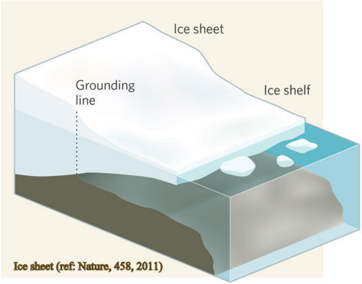



The trace of the log-posterior against MCMC iterations. From top to bottom: $\epsilon = 10^{-1}, 10^{-2}, 10^{-3}$. The red and black lines indicate FOM evaluations, where red means a rejected proposal, and black means an accepted proposal.

# Example 2: Arolla Glacier

Goal: estimating basal sliding coefficients from surface velocity measurements.

$$
\begin{aligned}
-\nabla \cdot [2\eta(\boldsymbol{u})\,\dot{\varepsilon}_{\boldsymbol{u}} - \boldsymbol{I}p] &= \rho \boldsymbol{g} && \text{in } \Omega \\
\nabla \cdot \boldsymbol{u} &= 0 && \text{in } \Omega \\
\boldsymbol{\sigma}_{\boldsymbol{u}} \boldsymbol{n} &= \boldsymbol{0} && \text{on } \Gamma_t \\
\boldsymbol{u} \cdot \boldsymbol{n} &= 0 && \text{on } \Gamma_b \\
\boldsymbol{T}\boldsymbol{\sigma}_{\boldsymbol{u}}\boldsymbol{n} + \exp(x)\boldsymbol{T}\boldsymbol{u} &= \boldsymbol{0} && \text{on } \Gamma_b
\end{aligned}
$$




- $\boldsymbol{u}$ ice flow velocity, $p$ pressure
- $\boldsymbol{\sigma}_{\boldsymbol{u}} = -\boldsymbol{I}p + 2\eta(\boldsymbol{u})\dot{\varepsilon}_{\boldsymbol{u}}$ stress tensor
- $\dot{\varepsilon}_{\boldsymbol{u}} = \frac{1}{2}(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^\top)$ strain rate tensor
- $\eta(\boldsymbol{u}) = \frac{1}{2} A^{-\frac{1}{n}} \dot{\varepsilon}_{\mathrm{II}}^{\frac{1-n}{2n}}$ effective viscosity
- $\dot{\varepsilon}_{\mathrm{II}} = \frac{1}{2}\mathrm{tr}(\dot{\varepsilon}_{\boldsymbol{u}}^2)$ second invariant of the strain rate tensor
- $\rho$ density, $\boldsymbol{g}$ gravity
- $\boldsymbol{n}$ unit normal vector
- $x$ log basal sliding coefficient
- $\boldsymbol{T} = \boldsymbol{I} - \boldsymbol{n} \otimes \boldsymbol{n}$ tangential operator
- $\Gamma_t$ and $\Gamma_b$ top and base boundaries

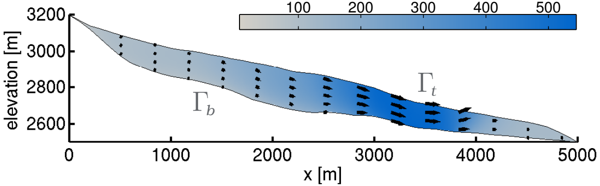
Joint work with Petra, Peherstorfer, Ghattas, Marzouk and Willcox

# Example 2: Arolla Glacier

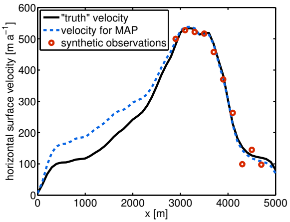
- Discretization system:

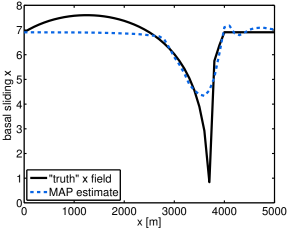
$$\boldsymbol{K}(\underline{\boldsymbol{u}},\underline{x})\underline{\boldsymbol{u}} + \boldsymbol{B}^\top \underline{p} = -\vec{\underline{r}}(\underline{\boldsymbol{u}},\underline{p}), \quad \boldsymbol{B}\underline{\boldsymbol{u}} = \boldsymbol{0},$$

  where $\boldsymbol{B}$ is the discretization of the divergence operator.
- One dimensional model to validate our methods

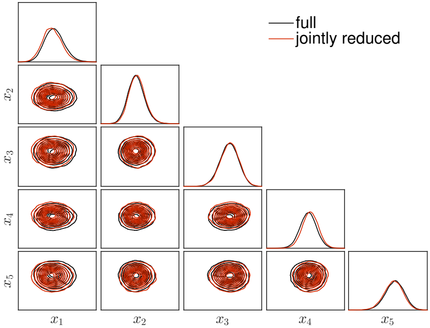


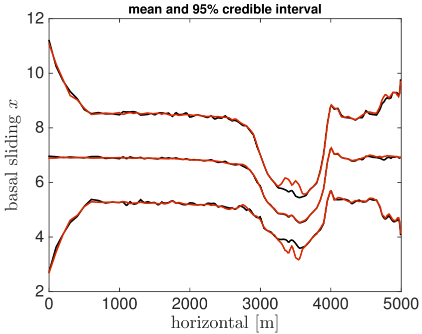

- Synthetic data and MAP estimate (used as the initial guess)



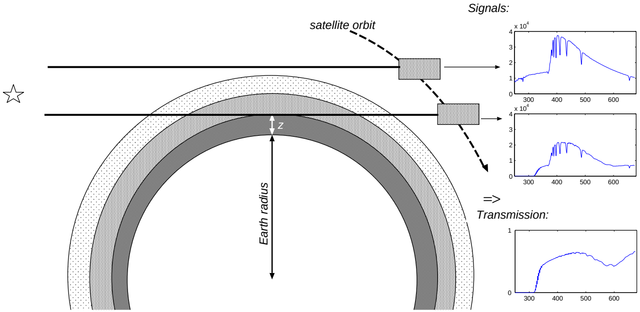



# Example 2: Arolla Glacier




- Full posterior: $139$ dimensional parameters $+$ $5373$ dimensional states
- Reduced: $50$ dim. states (also need parameter reduction, not discussed)
- Left: samples projected onto 5 leading parameter basis vectors
- Right: estimated parameter mean and credible intervals.

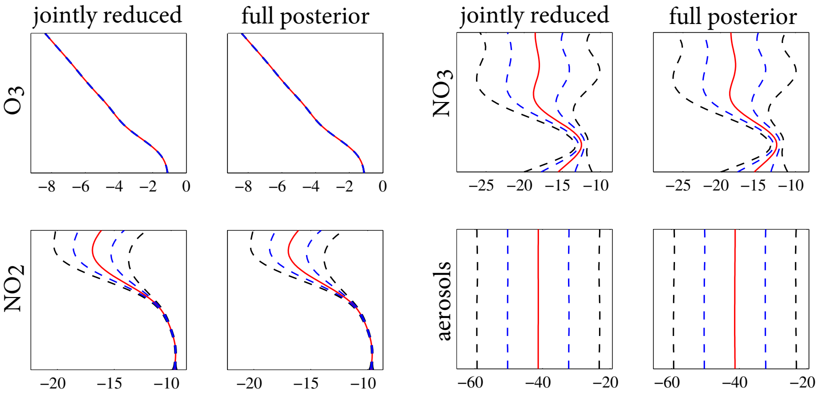



$$\text{Beer's law} \quad T_{\lambda,l} = \exp\left( -\int_l \sum_{\text{gas}} \alpha_\lambda^{\text{gas}}(h)\rho^{\text{gas}}(h)dh \right)$$

- Global Ozone Monitoring using Occultation Stars (GOMOS)
- Estimate gas densities $\rho^{\text{gas}}(h)$ from transmission spectrum $T_{\lambda,l}$
- Forward model is a nonlinear function $y = F(x)$, $F : \mathbb{R}^{200} \to \mathbb{R}^{70800}$

Joint work with Laine and Haario

# Example 3: GOMOS Remote Sensing




- Estimated gas density profiles
- Full posterior: $70800$ dimensional states / data
- Reduced: $45$ dim. states / data

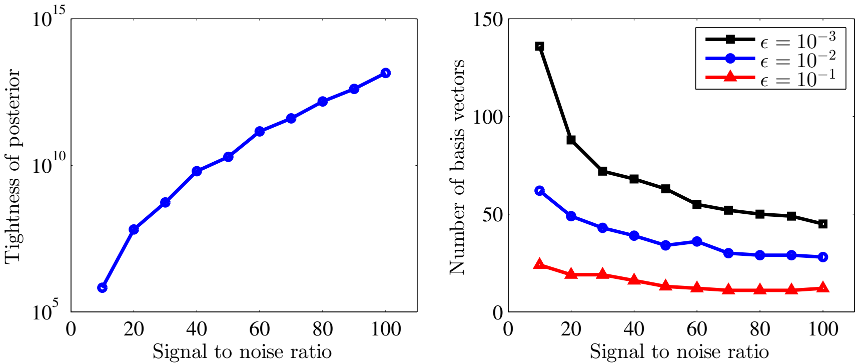
# Discussion

- We use **online adaptation** to construct effective reduced order models for accelerating Bayesian inverse problems
- Two algorithms are introduced, the **exact** delayed acceptance and the approximation based solely on ROM and error indicators.

Future works:

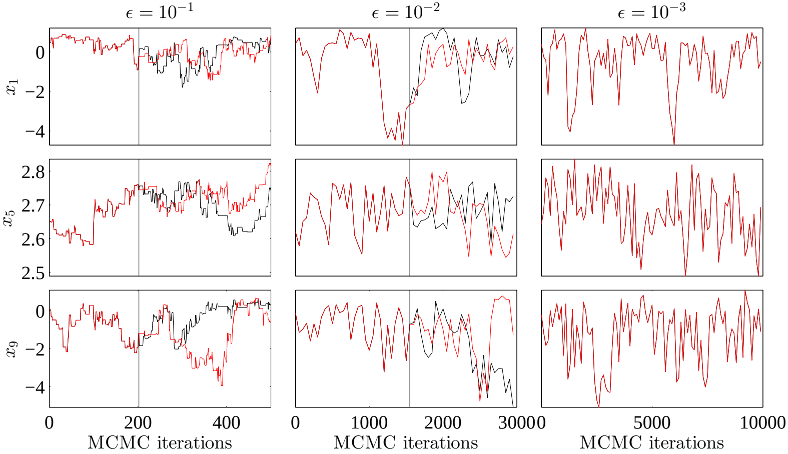
- How to use error estimators (bounds)?
- Use other surrogate modelling tools, e.g., tensor-train, sparse grids or low-discrepanc sequences.
- Sequential inference / data-assimilation.
- Exact MCMC using the approximation (randomisation techniques)

# A 9D Test Case: Influence of Data




- Influence of data is controlled by signal to noise ratio.
- The tightness of the posterior is $\prod_{i=1}^{N_p} \frac{\sigma_0(x_i)}{\sigma(x_i)}$.

# A 9D Test Case: Coupling Time




Coupling time between the MH algorithm sampling the approximate posterior and the MH sampling the exact posterior. From left to right, the approximate posterior uses ROM that constructed with different error threshold, $\epsilon = 10^{-1}, 10^{-2}, 10^{-3}$.

# GOMOS: vs. Prior Reduction

Comparison of marginals: