Nonlinear Model Reduction for Uncertainty Quantification in Large-Scale Inverse Problems

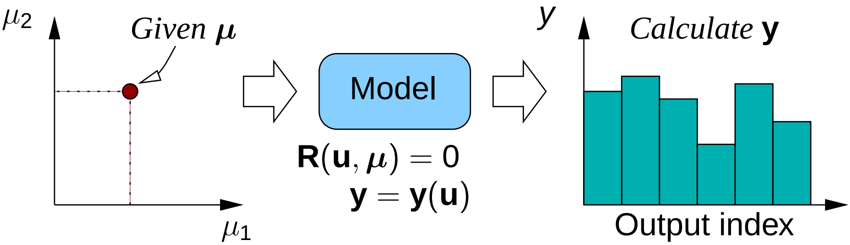
Krzysztof Fidkowski, David Galbally*, Karen Willcox*
(*MIT)

Computational Aerospace Sciences Seminar
Aerospace Engineering Department
University of Michigan

October 3, 2008

Outline

Forward Problem

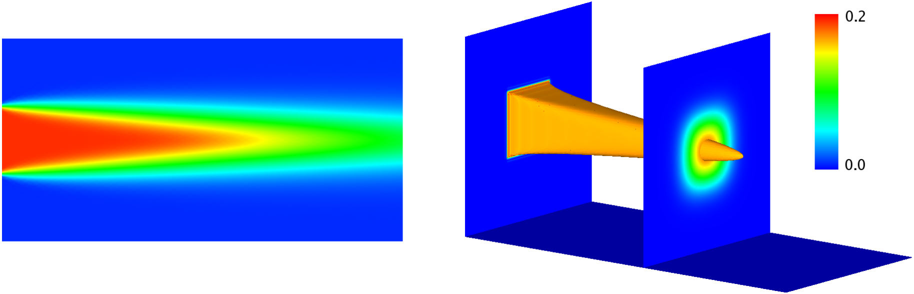


$\boldsymbol{\mu}$	=	parameter vector $[\mu_1, \mu_2]$
$\mathbf{y}$	=	output vector
$\mathbf{u}$	=	state vector
$\mathbf{R}(\mathbf{u}, \boldsymbol{\mu})$	=	model equations

Example

Model: Finite element discretization of a scalar convection-diffusion-reaction equation; scalar = fuel concentration

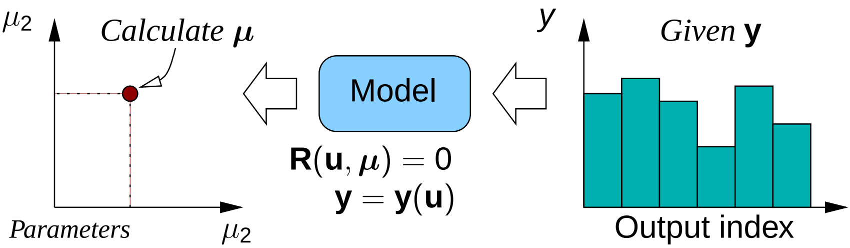
Sample fuel concentration contours



μ	=	reaction rate parameters
$\mathbf{y}$	=	average fuel concentrations at cut-planes
$\mathbf{u}$	=	finite element solution for fuel concentration

Inverse Problem

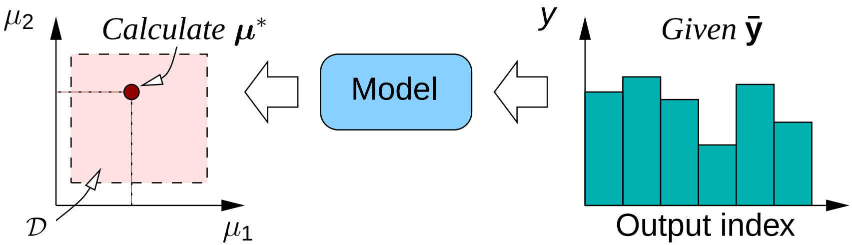
Given fuel concentrations, determine reaction rate parameters



Other applications:

- Medical imaging
- Circuit identification
- Model fitting
- Geophysics

Deterministic Inverse Solution



Determine the “best” value of the parameter vector:

$$\boldsymbol{\mu}^* = \arg\min_{\boldsymbol{\mu}} \|\mathbf{y}(\boldsymbol{\mu}) - \bar{\mathbf{y}}\|_2 \qquad \text{(minimization problem)}$$

$$\text{subject to} \quad \mathbf{R}(\mathbf{u};\boldsymbol{\mu}) = 0, \qquad \text{(model equations)}$$

$$\mathbf{y}(\boldsymbol{\mu}) \equiv \mathbf{y}(\mathbf{u}(\boldsymbol{\mu})),$$

$$\boldsymbol{\mu} \in \mathcal{D}. \qquad \text{(a priori knowledge)}$$

Deterministic Inverse Solution (ctd.)

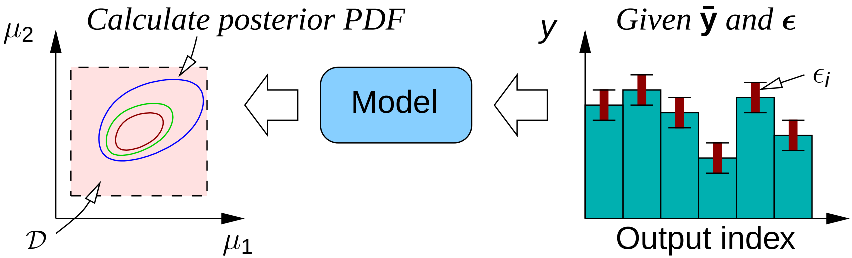
Shortcomings:

- Experimental errors not included
- No uncertainty quantification for the best estimate $\boldsymbol{\mu}^*$
- The inverse problem may be ill-posed:
  - no unique solution $\boldsymbol{\mu}^*$
  - $\boldsymbol{\mu}^*$ sensitive to small perturbations in $\bar{\mathbf{y}}$

Practical solution is some form of **regularization**, for example:

$$
\begin{aligned}
\boldsymbol{\mu}^* &= \arg\min_{\boldsymbol{\mu}} \|\mathbf{y}(\boldsymbol{\mu}) - \bar{\mathbf{y}}\|_2 + \beta\|\boldsymbol{\mu}\|_2 \\
\beta &= \text{regularization parameter}
\end{aligned}
$$

Statistical Inverse Solution



For example, $\boldsymbol{\epsilon}$ = normally-distributed measurement errors, each with standard deviation σ.

With this measurement error, **likelihood function** is:

$$p(\bar{\mathbf{y}}|\boldsymbol{\mu}) \propto \exp\left[-\frac{1}{2\sigma^2}(\bar{\mathbf{y}} - \mathbf{y}(\boldsymbol{\mu}))^T(\bar{\mathbf{y}} - \mathbf{y}(\boldsymbol{\mu}))\right]$$

= probability of measuring $\bar{\mathbf{y}}$ given $\boldsymbol{\mu}$

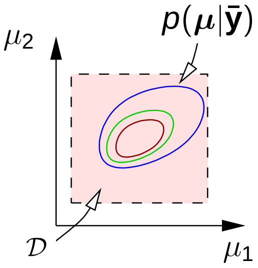
Posterior Probability Distribution

Using Bayes' theorem:

$$p(\boldsymbol{\mu}|\bar{\mathbf{y}}) = \frac{1}{p(\bar{\mathbf{y}})} p(\bar{\mathbf{y}}|\boldsymbol{\mu}) p(\boldsymbol{\mu}),$$

so that the **posterior PDF** is (assuming a uniform prior $p(\boldsymbol{\mu})$)

$$p(\boldsymbol{\mu}|\bar{\mathbf{y}}) \propto \begin{cases} \exp\left[-\frac{1}{2\sigma^2}(\bar{\mathbf{y}} - \mathbf{y}(\boldsymbol{\mu}))^T(\bar{\mathbf{y}} - \mathbf{y}(\boldsymbol{\mu}))\right], & \text{if } \boldsymbol{\mu} \in \mathcal{D} \\ 0, & \text{otherwise.} \end{cases}$$



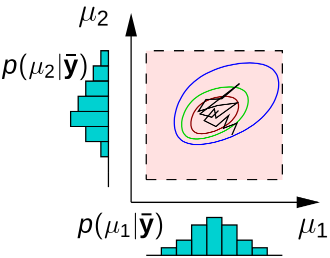
The posterior PDF is *inferred* from the measured outputs, $\bar{\mathbf{y}}$, and a model for the measurement error, σ, using $\mathbf{y}(\boldsymbol{\mu})$.

How do we describe $p(\boldsymbol{\mu}|\bar{\mathbf{y}})$?

MCMC Sampling

Use Markov Chain Monte Carlo (MCMC) to *sample* the posterior PDF

- Take a random walk in parameter space
- Generate sequence: $\boldsymbol{\mu}^1, \boldsymbol{\mu}^2, \ldots$

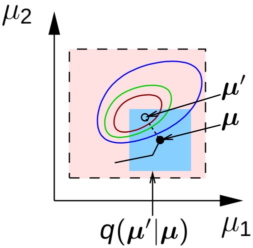


Taking a step given $\boldsymbol{\mu} = \boldsymbol{\mu}^i$:

- Pick $\boldsymbol{\mu}'$ from a proposal distribution, q
- Accept $\boldsymbol{\mu}'$ (i.e. $\boldsymbol{\mu}^{i+1} = \boldsymbol{\mu}'$) with probability:

$$\alpha(\boldsymbol{\mu}'|\boldsymbol{\mu}) = \min\left[1, \frac{p(\boldsymbol{\mu}'|\bar{\mathbf{y}})q(\boldsymbol{\mu}'|\boldsymbol{\mu})}{p(\boldsymbol{\mu}|\bar{\mathbf{y}})q(\boldsymbol{\mu}|\boldsymbol{\mu}')}\right],$$

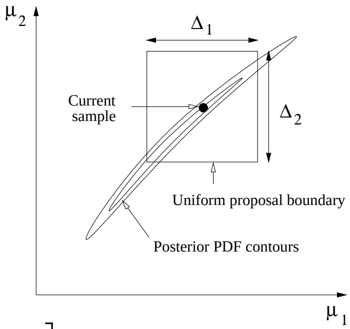
Otherwise reject it: ($\boldsymbol{\mu}^{i+1} = \boldsymbol{\mu}$)



The Proposal Distribution, $q(\boldsymbol{\mu}'|\boldsymbol{\mu})$

Choice of $q(\boldsymbol{\mu}'|\boldsymbol{\mu})$ governs exploration of the parameter space, and affects the acceptance probability α.

1. Uniform box
   - Size $\boldsymbol{\Delta} = [\Delta_1, \Delta_2]$ centered at $\boldsymbol{\mu}$
   - $q(\boldsymbol{\mu}'|\boldsymbol{\mu}) = q(\boldsymbol{\mu}|\boldsymbol{\mu}') = \text{const.}$
   - Inefficient for anisotropic posteriors



2. Stretched ellipse

$$q(\boldsymbol{\mu}'|\boldsymbol{\mu}) \propto \exp\left[-\frac{1}{2\delta^2}(\boldsymbol{\mu}'-\boldsymbol{\mu})^T\mathbf{H}(\boldsymbol{\mu})(\boldsymbol{\mu}'-\boldsymbol{\mu})\right]$$

$$\mathbf{H}(\boldsymbol{\mu}) = \frac{1}{2\sigma^2}\left[\frac{\partial\mathbf{y}}{\partial\boldsymbol{\mu}}(\boldsymbol{\mu})\right]^T\frac{\partial\mathbf{y}}{\partial\boldsymbol{\mu}}(\boldsymbol{\mu})$$

δ = step-size param

$\frac{\partial\mathbf{y}}{\partial\boldsymbol{\mu}}$ via FD

Sampling Statistics

N_m = number of MCMC samples

- $\boldsymbol{\mu}^i$, $i = 1..N_m$ are drawn from the posterior probability distribution
- for $N_m \to \infty$, expect convergence to the actual probability distribution of $\boldsymbol{\mu}^*$
- for finite N_m, can only *estimate* statistics

Estimator of a statistical quantity:

$$\bar{f} \equiv \frac{1}{N_m} \sum_{i=1}^{N_m} g(\boldsymbol{\mu}^i)$$

mean of μ_j	$g(\boldsymbol{\mu}) = \mu_j$
variance of μ_j	$g(\boldsymbol{\mu}) = (\mu_j - \bar{\mu}_j)^2$
	($\bar{\mu}_j$ = sample mean)

- N_m is usually tens of thousands
- Each evaluation of acceptance probability requires a forward run
- **Cost becomes prohibitive for large simulations**

Model Reduction

Goals:

- Create a computationally inexpensive emulator of the forward simulation
- Require accuracy for $\boldsymbol{\mu} \in \mathcal{D}$
- Retain physics of the problem
- Take into account non-linearities

Assumption

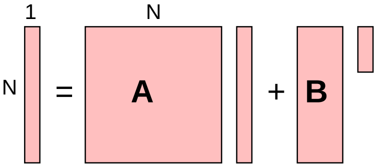
For $\boldsymbol{\mu} \in \mathcal{D}$, solution $\mathbf{u}$ resides in a low dimensional manifold – i.e. can represent it well using $n \ll N$ degrees of freedom.

Model Reduction Using Linear Projection

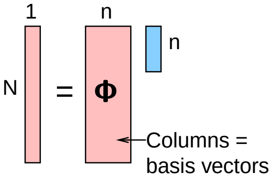
N = # unknowns in full system, $(\approx$ millions)

n = # unknowns in reduced system, (≈ 100)

$$\dot{\mathbf{u}} = \mathbf{A}\mathbf{u} + \mathbf{B}\mu$$

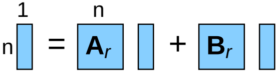


$$\mathbf{u} \approx \mathbf{\Phi}\mathbf{u_r}, \quad \mathbf{\Phi}^T\mathbf{\Phi} = \mathbf{I}^{n\times n}$$



Multiply original system by $\mathbf{\Phi}^T$ to obtain the **reduced system**:

$$\dot{\mathbf{u}}_r = \underbrace{\mathbf{\Phi}^T\mathbf{A}\mathbf{\Phi}}_{\mathbf{A}_r}\mathbf{u}_r + \underbrace{\mathbf{\Phi}^T\mathbf{B}}_{\mathbf{B}_r}\mu$$



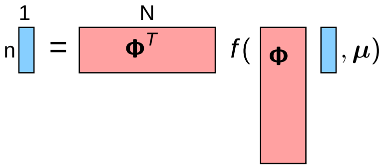
- Can precompute $\mathbf{A}_r$ and $\mathbf{B}_r$
- System is of size n
- No order N operations to run the reduced system

Model Reduction for Nonlinear Systems

$$\dot{\mathbf{u}} = f(\mathbf{u}, \boldsymbol{\mu}), \quad f(\cdot, \boldsymbol{\mu}) = \text{nonlinear function}$$

Multiplying the original system by $\mathbf{\Phi}^T$ we obtain a “reduced system”:

$$\dot{\mathbf{u}}_r = \mathbf{\Phi}^T f(\mathbf{u}, \boldsymbol{\mu})$$

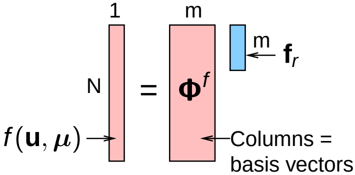


- n unknowns in reduces system but ...
- Cannot precompute any matrix products because of f
- Need N nonlinearity evaluations – this will dominate the cost!

Nonlinearity Expansion

Key assumption: $f(\mathbf{u},\boldsymbol{\mu})$ resides in a low manifold of dimension $m \approx n$.

$$f(\mathbf{u},\boldsymbol{\mu}) \approx \boldsymbol{\Phi}^f \mathbf{f}_r, \qquad \mathbf{f}_r \in \mathbb{R}^m$$



Substituting into the reduced system:

$$\dot{\mathbf{u}}_r = \underbrace{\boldsymbol{\Phi}^T \boldsymbol{\Phi}^f}_{n \times m} \mathbf{f}_r \qquad (N \text{ not present})$$

But, evaluating $\mathbf{f}_r$ directly still involves N:

$$\mathbf{f}_r = \underbrace{(\boldsymbol{\Phi}^f)^T}_{m \times N} \underbrace{f(\mathbf{u},\boldsymbol{\mu})}_{N \times 1} \qquad (\text{order } N \text{ dependent})$$

Masked Projection

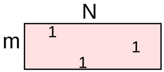
Compute $\mathbf{f}_r$ = nonlinearity expansion coefficients *approximately*:

$$
\begin{aligned}
f(\mathbf{u}, \boldsymbol{\mu}) &\approx \boldsymbol{\Phi}^f \mathbf{f}_r \\
\mathbf{Z} f(\mathbf{u}, \boldsymbol{\mu}) &\approx \mathbf{Z}\boldsymbol{\Phi}^f \mathbf{f}_r \\
\mathbf{f}_r &\approx (\mathbf{Z}\boldsymbol{\Phi}^f)^{-1} \mathbf{Z} f(\mathbf{u}, \boldsymbol{\mu}) \\
\Rightarrow f(\mathbf{u}, \boldsymbol{\mu}) &\approx \underbrace{\boldsymbol{\Phi}^f (\mathbf{Z}\boldsymbol{\Phi}^f)^{-1}}_{\boldsymbol{\Psi} \in \mathbb{R}^{N \times m}} \mathbf{Z} f(\mathbf{u}, \boldsymbol{\mu})
\end{aligned}
$$

$\mathbf{Z}$ is an $m \times N$ **mask** matrix

- Mostly zeros
- Ones in columns where f is to be evaluated

$\mathbf{Z} =$

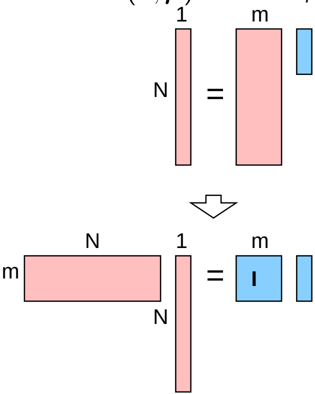


- $\boldsymbol{\Psi} = \boldsymbol{\Phi}^f (\mathbf{Z}\boldsymbol{\Phi}^f)^{-1}$ can be precomputed
- $\mathbf{Z} f(\mathbf{u}, \boldsymbol{\mu})$ consists of m evaluations of the nonlinearity
- Similar to gappy POD [Everson Sirovicz, 1995]

Comparison to Direct Projection

Direct Projection

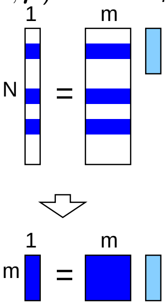
$$f(\mathbf{u},\mu) = \boldsymbol{\Phi}^f \, \mathbf{f}_r$$



$$(\boldsymbol{\Phi}^f)^T f(\mathbf{u},\mu) = \mathbf{f}_r$$

Masked Projection

$$\mathbf{Z}f(\mathbf{u},\mu) \approx \mathbf{Z}\boldsymbol{\Phi}^f \, \mathbf{f}_r$$

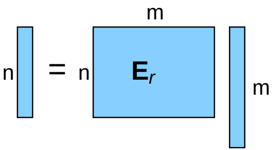


$$\mathbf{Z}f(\mathbf{u},\mu) \approx \mathbf{Z}\boldsymbol{\Phi}^f \, \mathbf{f}_r$$

Reduced Nonlinear System

Using $f(\mathbf{u}, \mu) \approx \mathbf{\Psi}^f f(\mathbf{Z}\mathbf{u}, \mu)$,

$$
\begin{aligned}
\dot{\mathbf{u}}_r &= \mathbf{\Phi}^T f(\mathbf{u}, \mu) \\
&\approx \underbrace{\mathbf{\Phi}^T \mathbf{\Psi}^f}_{\mathbf{E}_r \in \mathbb{R}^{n \times m}} f(\mathbf{Z}\mathbf{u}, \mu)
\end{aligned}
$$



Steps

- Form $\mathbf{\Phi}$ and $\mathbf{\Psi}^f$ basis matrices by, for example, POD on a set of snapshots
- Choose a mask $\mathbf{Z}$ and calculate $\mathbf{\Psi} = \mathbf{\Phi}^f (\mathbf{Z}\mathbf{\Phi}^f)^{-1}$
- Calculate $\mathbf{E}_r = \mathbf{\Phi}^T \mathbf{\Psi}^f$ offline
- Each forward solve of reduced model now involves only $m \approx n$ nonlinearity evaluations

Choosing a Mask, $\mathbf{Z}$

Accuracy of reduced model depends on $\mathbf{Z}$

- **Option 1**: Choose $\mathbf{Z}$ to minimize cond($\mathbf{Z}\mathbf{\Phi}^f$) [Willcox, 2006]
- **Option 2**: Choose $\mathbf{Z}$ to minimize error between the masked projection and the full projection of K snaphsots, ξ_k^f :

$$\mathbf{Z} = \arg\min_{\mathbf{Z}} \sum_{k=1}^{K} \|(\mathbf{\Phi}^f)^T \xi_k^f - (\mathbf{Z}\mathbf{\Phi}^f)^{-1} \mathbf{Z} \xi_k^f\|_2^2$$

  BPIM = Best Points Interpolation Method [Nguyen *et al*, 2007]

- **Option 3**: Choose $i+1$st mask point *recursively* as the index where the error between Φ_{i+1}^f and its reconstruction using the first i basis vectors is maximum.

  EIM = Empirical Interpolation Points Method [Nguyen *et al*, 2007]

Convection-Diffusion-Reaction Equations

u = scalar fuel concentration

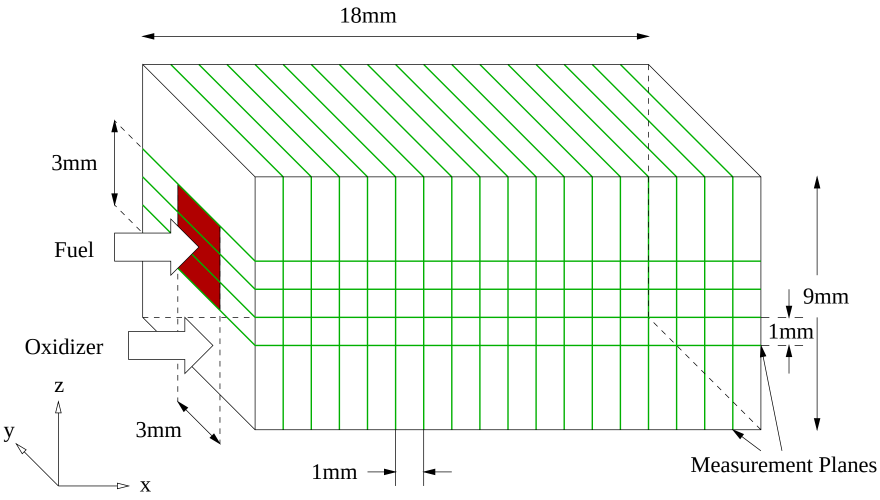
$$\begin{aligned} \nabla \cdot (\mathbf{U}u) - \nabla(\nu \nabla u) + f(u, \boldsymbol{\mu}) &= 0 && \text{in } \Omega, \\ u &= u_D && \text{on } \partial\Omega_D, \\ \nabla u \cdot \mathbf{n} &= 0 && \text{on } \partial\Omega \backslash \partial\Omega_D, \end{aligned}$$

$$\begin{aligned} u &= \text{fuel concentration} \\ \mathbf{U} &= \text{velocity (constant)} \\ \nu &= \text{diffusion coefficient (constant)} \end{aligned}$$

Nonlinear reaction term:

$$f(u, \boldsymbol{\mu}) = Au(c - u)e^{\frac{-E}{d-u}}, \quad \boldsymbol{\mu} = (\ln(A), E)$$

Combustor Model



Finite Element Discretization

- 2D: Streamwise Upwind Petrov Galerkin (SUPG)
- 3D: Discontinuous Galerkin (DG)

General discrete form (N unknowns, M nonlinearity evaluations):

$$\mathbf{R}(\mathbf{u};\boldsymbol{\mu}) = \mathbf{R}_0 + \mathbf{A}\mathbf{u} + \mathbf{E}\, f(\mathbf{D}\mathbf{u},\boldsymbol{\mu}) = 0$$

D interpolates **u** to M quadrature points
E sums up the nonlinear evaluations

Reduced model (n unknowns, m nonlinearity evaluations):

$$\mathbf{R}_r + \mathbf{A}_r\mathbf{u}_r + \mathbf{E}_r\, f(\mathbf{D}_r\mathbf{u}_r,\boldsymbol{\mu}) = 0$$

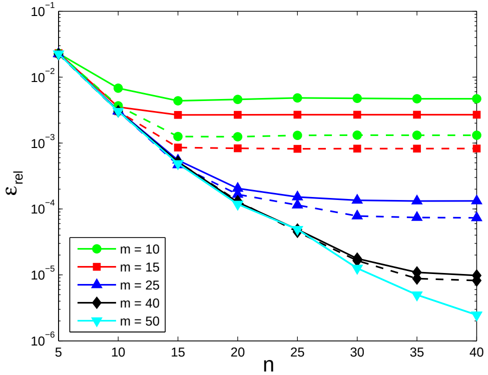
$$\begin{aligned} \mathbf{E}_r &= \boldsymbol{\Phi}^T\mathbf{E}\boldsymbol{\Psi} && \in \mathbb{R}^{n\times m} \\ \mathbf{D}_r &= \mathbf{Z}\mathbf{D}\boldsymbol{\Phi} && \in \mathbb{R}^{m\times n} \end{aligned}$$

2D Reduced Model Performance

- Basis constructed from $K = 196$ snapshots.
- Ξ^{test} = 23×23 test grid in parameter space

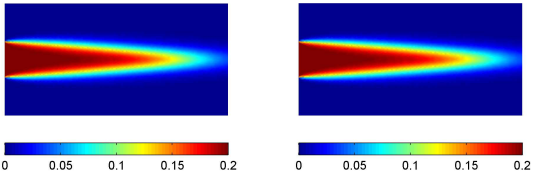
Average relative error:

$$\varepsilon_{\text{rel}} = \underset{\boldsymbol{\mu} \in \Xi^{\text{test}}}{\text{mean}} \frac{\|\boldsymbol{y}(\boldsymbol{\mu}) - \boldsymbol{y}_r(\boldsymbol{\mu})\|}{\|\boldsymbol{y}(\boldsymbol{\mu})\|}$$



n	m	ε_{rel}	Online time
5	50	2.25 E−2	1.59 E−5
10	50	3.03 E−3	1.61 E−5
20	50	1.18 E−4	1.63 E−5
30	50	1.26 E−5	1.71 E−5
40	50	2.47 E−6	2.00 E−5

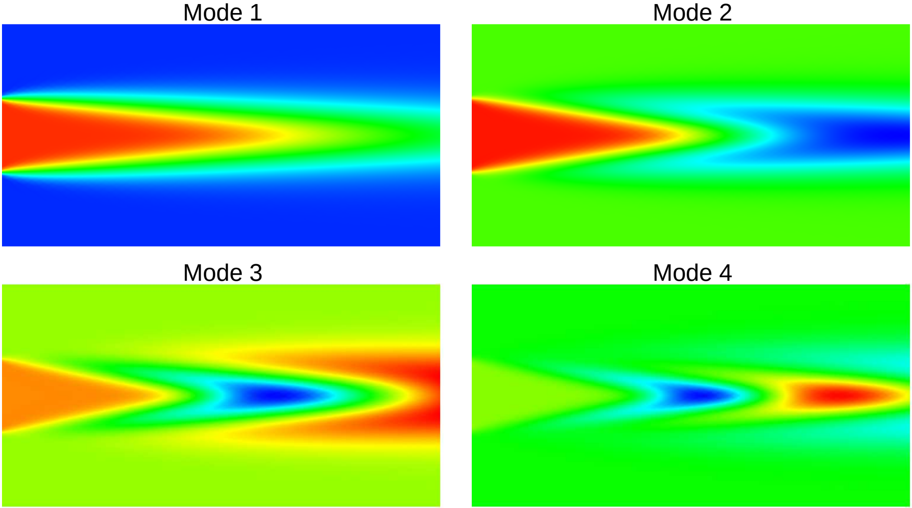
Online time is relative to FEM solution



$n = 40$, $m = 50$ field comparison with full order solution

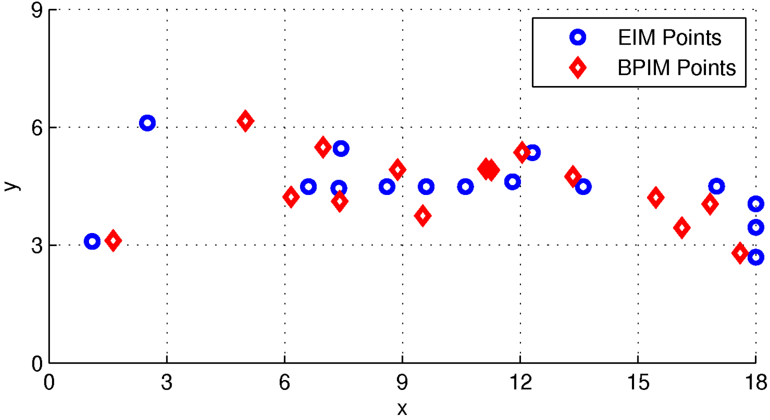
2D Reduced Model Basis Vectors

Obtained by Proper Orthogonal Decomposition (Karhunen Loève expansion) of 196 snapshots in a 14×14 grid in parameter space. First four modes:



2D Mask Points

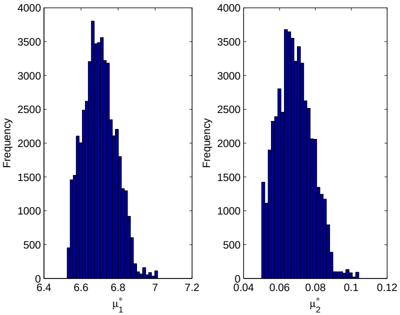
For $m = 15$, the mask $\mathbf{Z} \in \mathbb{R}^{m \times N}$ contains 15 nonzero entries. These are the points at which the nonlinear term is evaluated.

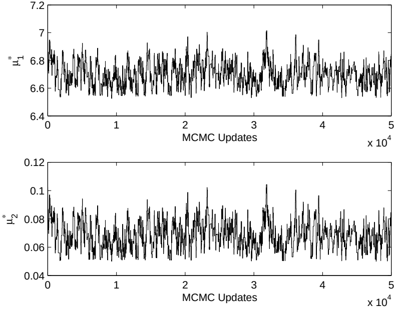


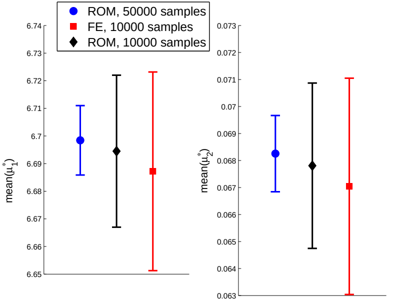
[Galbally 2005]

2D Inverse Problem Results

- Generated $\bar{\mathbf{y}}$ with FEM model and $\sigma = 1.5\%$ measurement error
- Goal: determine PDF of parameters $\mu_1^* = \log(A^*)$ and $\mu_2^* = E^*$
- Constructed a Markov chain of size $N_m = 50,000$ with the $n = 40$, $m = 50$ reduced model

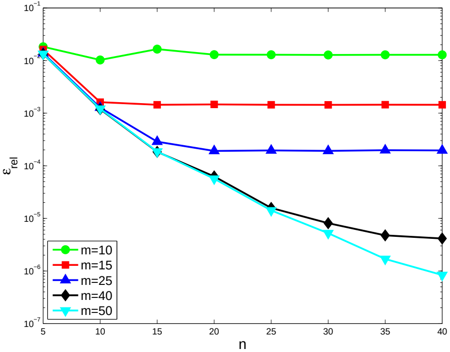




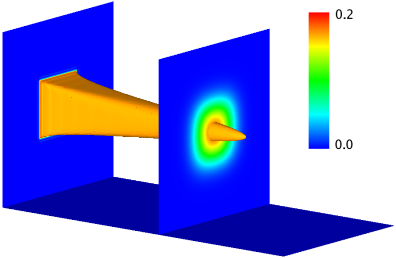


3D Reduced Model Performance

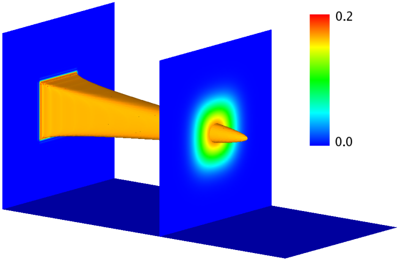
- Full-order FEM model: 8.5 million unknowns (13h CPU time)
- Basis constructed from $K = 169$ snapshots.
- $\Xi^{\text{test}} = 17 \times 17$ test grid in parameter space
- Reduced model: .1s CPU time



Finite element sol. (8.5 million unknowns)



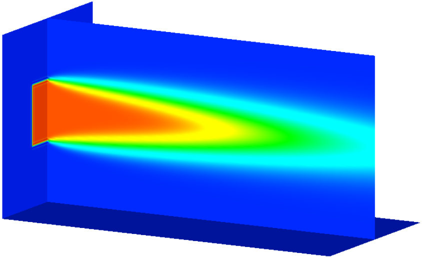
$n = 40$, $m = 50$ reduced model (EIM)



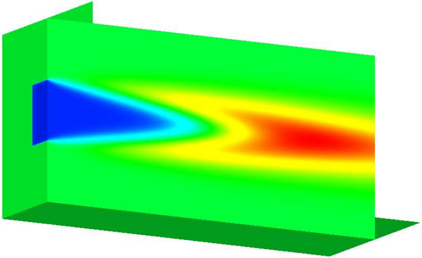
3D Reduced Model Basis Vectors

POD of 169 snapshots in a 13×13 grid in parameter space. First four modes:

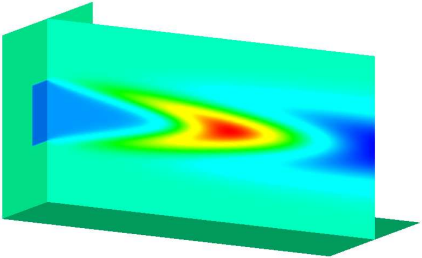
Mode 1



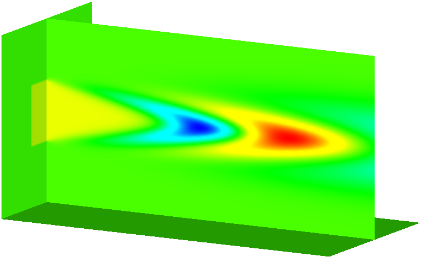
Mode 2



Mode 3

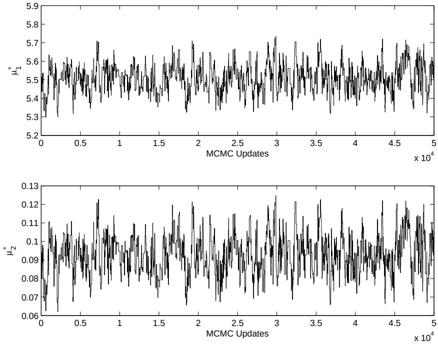


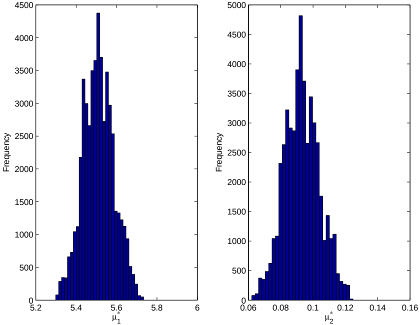
Mode 4



3D Inverse Problem Results

- Generated $\bar{\mathbf{y}}$ with FEM model and $\sigma = 1.5\%$ measurement error
- Goal: determine PDF of parameters $\mu_1^* = \log(A^*)$ and $\mu_2^* = E^*$
- Constructed a Markov chain of size $N_m = 50{,}000$ with the $n = 40$, $m = 50$ reduced model

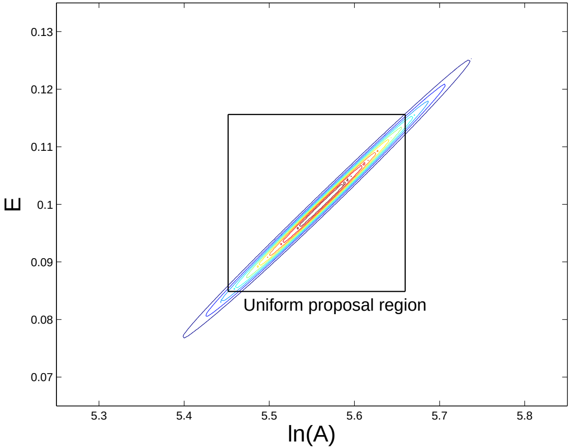




- Uniform proposal distribution
- Acceptance rate = 3.4% (low)

Posterior Anisotropy

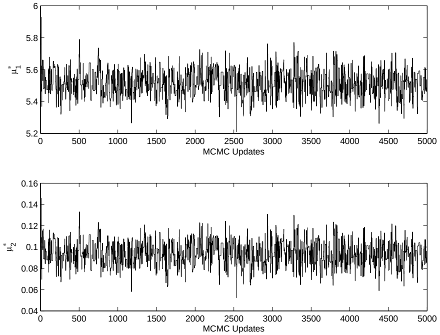
Low acceptance rate of uniform proposal attributed to anisotropy in the posterior PDF, $p(\mathbf{y}|\boldsymbol{\mu})$:

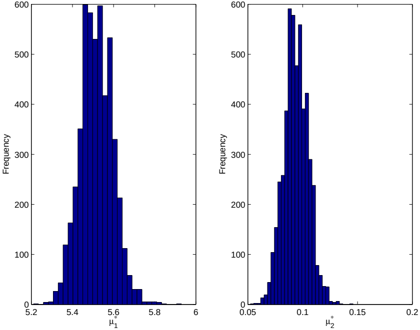


Improve acceptance rate via a stretched-Gaussian proposal

3D Inverse Problem Results, Stretched Proposal

- Used $\delta = 1.5$ for the dimensionless step-size parameter
- Finite differencing for $\partial \mathbf{y} / \partial \boldsymbol{\mu}$
- Acceptence rate now 25%
- Only needed $N_m = 5,000$ samples for same statistics





Note: MCMC runs with full-order model are prohibitive (13h CPU per forward solve)

Summary and Conclusions

- Presented a nonlinear model reduction technique in a projection framework
- Built on previous work in gappy POD, missing point estimation, masked projection, coefficient-function approximation
- Applied reduction to a parameter estimation problem in a Bayesian inference setting
- Vast speedup of reduced model makes such an inverse problem solution possible
- Additional work:
  - Model-constrained adaptive sampling to generate snapshots as number of parameters is increased
  - Quantification of model reduction errors on statistics of interest