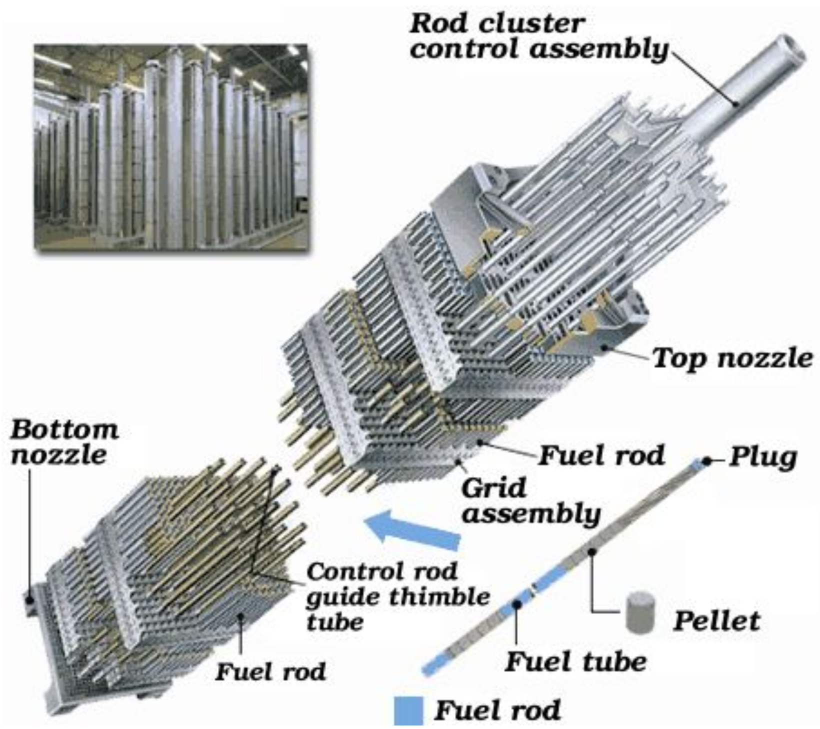
Pressurized Water Reactors (PWR)

3-D Neutron Transport Equations:

$$\frac{1}{|v|}\frac{\partial \varphi}{\partial t} + \Omega \cdot \nabla \varphi + \Sigma_t(r,E)\varphi(r,E,\Omega,t)$$
$$= \int_{4\pi} d\Omega' \int_0^\infty dE' \Sigma_s(E' \to E, \Omega' \to \Omega)\varphi(r,E',\Omega',t)$$
$$+ \underline{\frac{\chi(E)}{4\pi}} \int_{4\pi} d\Omega' \int_0^\infty dE' \underline{\nu(E')}\,\underline{\Sigma_f(E')}\varphi(r,E',\Omega',t)$$



Challenges:

• Linear in the state but function of 7 independent variables:

$$r = x, y, z; E; \Omega = \theta, \phi; t$$

• Very large number of inputs; e.g., 100,000; Active subspace construction is critical.

• ORNL Code SCALE: can take minutes to hours to run.

• SCALE TRITON has adjoint capabilities via TSUNAMI-2D and NEWT.

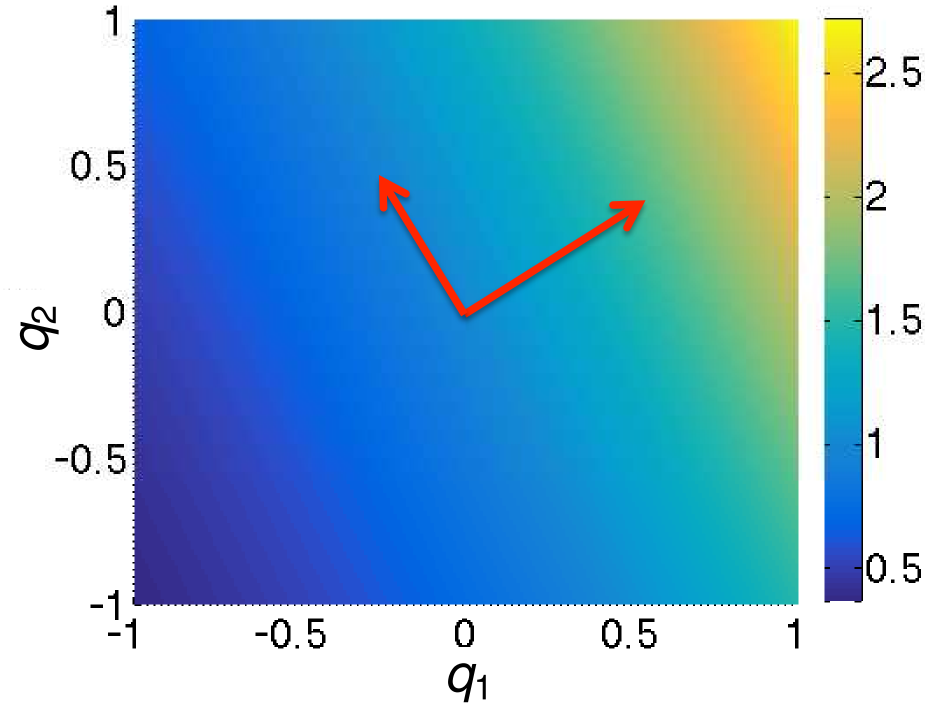
Active Subspaces

Note:

- Functions may vary significantly in only a few directions
- "Active" directions may be linear combination of inputs

Example: $y = \exp(0.7q_1 + 0.3q_2)$

- Varies most in [0.7, 0.3] direction
- No variation in orthogonal direction



A Bit of History:

- Often attributed to Russi (2010).
- Concept same as *identifiable subspaces* from systems and control; e.g., Reid (1977).
- For linearly parameterized problems, active subspace given by SVD or QR; Beltrami (1873), Jordan (1874), Sylvester (1889), Schmidt (1907), Weyl (1912). See 1993 *SIAM Review* paper by Stewart.

Active Subspaces

Note: Sensitivity analysis isolate *subsets* of influential parameters but ineffective for *subspaces* that are not aligned with coordinate axes.

Linearly Parameterized Problems: $y = Aq \quad , \; y \in \mathbb{R}^n \quad , \; q \in \mathbb{R}^p \quad , \; A$ is $n \times p$

Example: $y_i = q_2 x_i \; , \; i = 1, 2, 3$

$$q = [q_1, q_2]$$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 0 & x_1 \\ 0 & x_2 \\ 0 & x_3 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}$$

Here

$$NI(q) = \mathcal{N}(A) = c \begin{bmatrix} 1 \\ 0 \end{bmatrix} \; , \; c \in \mathbb{R}$$

$$I(q) = \mathcal{R}(A^T) = c \begin{bmatrix} 0 \\ 1 \end{bmatrix} \; , \; c \in \mathbb{R}$$

Null space of A

$$\mathcal{N}(A) = \{q \in \mathbb{R}^p \,|\, Aq = 0\}$$

Range

$$\mathcal{R}(A^T) = \{b \in \mathbb{R}^p \,|\, b = A^T z \text{ for some } z \in \mathbb{R}^n\}$$

Note: $\mathcal{N}(A^T A) = \mathcal{N}(A) \; , \; \mathcal{R}(A^T A) = \mathcal{R}(A^T)$

Good Reference: Ilse C.F. Ipsen, *Numerical Matrix Analysis*, SIAM, 2009

Active Subspaces

Example: $y = [2 \quad 1] \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}$

Here

$$NI(q) = \mathcal{N}(A) = c \begin{bmatrix} -\frac{1}{2} \\ 1 \end{bmatrix}, \; c \in \mathbb{R}$$

$$I(q) = \mathcal{R}(A^T) = c \begin{bmatrix} 2 \\ 1 \end{bmatrix}, \; c \in \mathbb{R}.$$

Deterministic Algorithms

Linearly Parameterized Problems: $y = Aq \quad , \ y \in \mathbb{R}^n \quad , \ q \in \mathbb{R}^p \quad , \ A$ is $n \times p$

Singular Value Decomposition (SVD):

$$A = U\Sigma V^T \quad , \ \Sigma = [S \quad 0]$$

$$S = \begin{bmatrix} \sigma_1 & & & \\ & \ddots & & \\ & & \sigma_r & \\ & & & 0 \end{bmatrix} \quad , \ \sigma_1 \geqslant \sigma_2 \geqslant \cdots \geqslant \sigma_r \geqslant \varepsilon$$

and

$$U = [U_r \quad U_{n-r}] \ , \ U_r \in \mathbb{R}^{n \times r} \ , \ U_{n-r} \in \mathbb{R}^{n \times (n-r)}$$

$$V = [V_r \quad V_{p-r}] \ , \ V_r \in \mathbb{R}^{p \times r} \ , \ V_{p-r} \in \mathbb{R}^{p \times (p-r)}$$

Rank Revealing QR Decomposition: $A^T P = QR$

Problem: Neither is directly applicable when n or p are very large; e.g., millions.

Solution: Random range finding algorithms.

Random Range Finding Algorithms: Linear Problems

Algorithm: Halko, Martinsson and Tropp, SIAM Review, 2011

1. Choose ℓ random inputs q^i and compute outputs $y^i = Aq^i$ which are compiled in the $m \times \ell$ matrix Y.

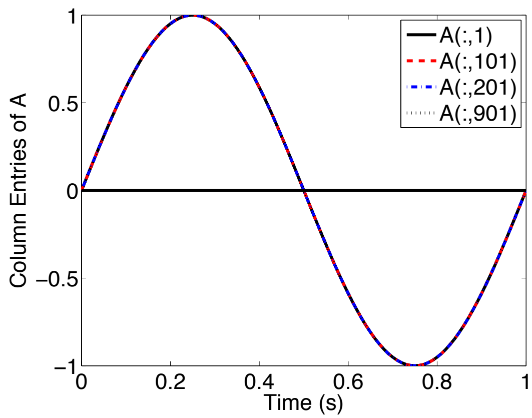
2. Take a pivoted QR factorization $Y = QR$ to construct a matrix Q whose columns form an orthonormal basis for the range of Y.

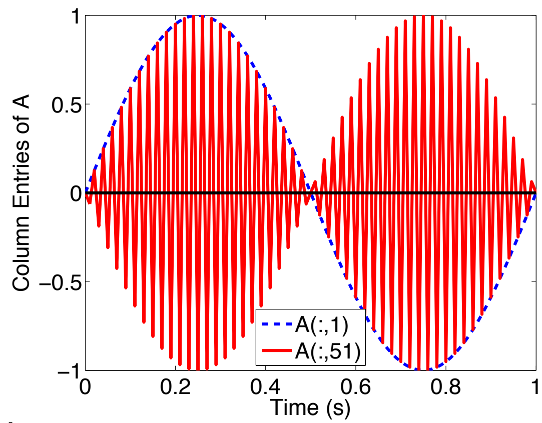
Example:

$$\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} \sin(2\pi t_1) & \cdots & \sin(2\pi p t_1) \\ \vdots & & \vdots \\ \sin(2\pi t_n) & \cdots & \sin(2\pi p t_n) \end{bmatrix} \begin{bmatrix} q_1 \\ \vdots \\ q_p \end{bmatrix}$$

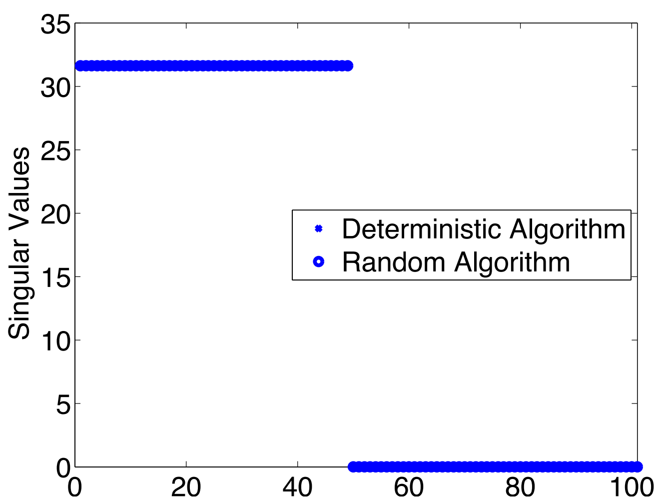
Random Range Finding Algorithms: Linear Problems

Example: m = 101, p = 1000: Analytic value for rank is 49

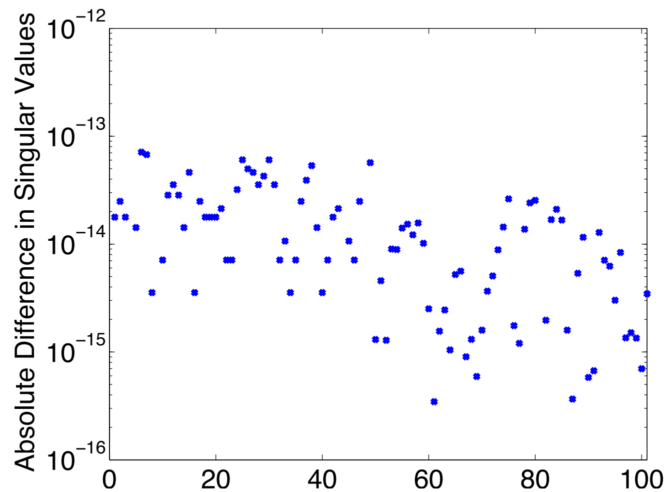




Aliasing



Singular Values



Absolute Difference Between Singular Values

Example: m = 101, p = 1,000,000: Random algorithm still viable

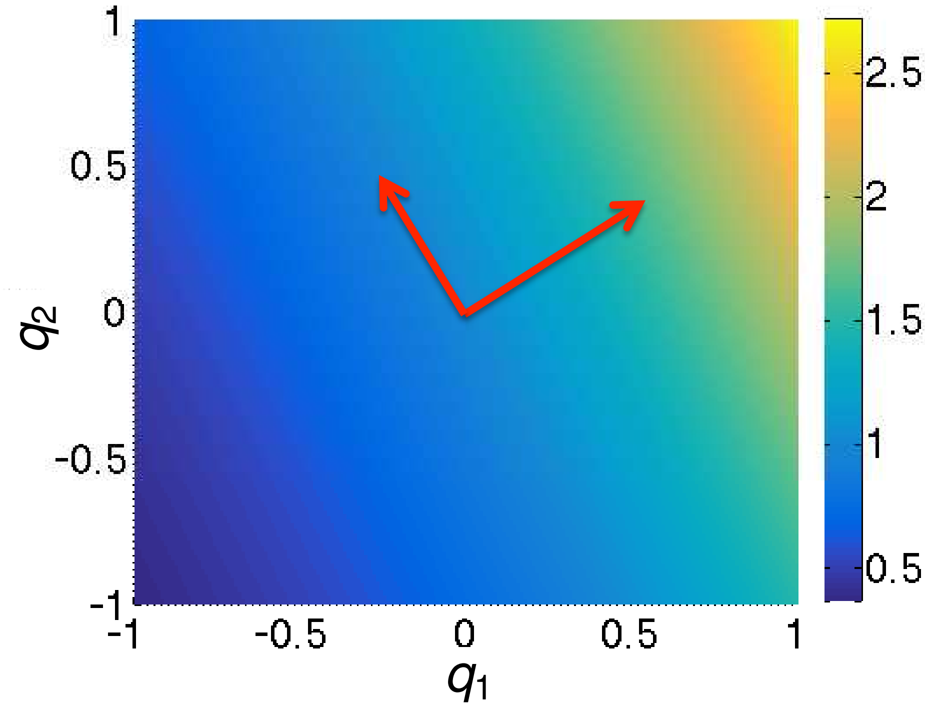
Active Subspaces for Nonlinearly Parameterized Problems

Note:

- Functions may vary significantly in only a few directions
- "Active" directions may be linear combination of inputs

Example: $y = \exp(0.7q_1 + 0.3q_2)$

- Varies most in [0.7, 0.3] direction
- No variation in orthogonal direction



A Bit of History:

- Often attributed to Russi (2010).
- Concept same as *identifiable subspaces* from systems and control; e.g., Reid (1977).
- For linearly parameterized problems, active subspace given by SVD or QR; Beltrami (1873), Jordan (1874), Sylvester (1889), Schmidt (1907), Weyl (1912). See 1993 *SIAM Review* paper by Stewart.

Gradient-Based Active Subspace Construction

Active Subspace: Consider

$$f = f(q)\ , q \in \mathcal{Q} \subseteq \mathbb{R}^p$$

- E.g., see [Constantine, SIAM, 2015; Stoyanov & Webster, *IJUQ*, 2015]

and

$$\nabla_q f(q) = \left[\frac{\partial f}{\partial q_1}, \cdots, \frac{\partial f}{\partial q_p}\right]^T$$

Construct outer product

$$C = \int (\nabla_q f)(\nabla_q f)^T \rho dq$$

$\rho(q)$: Distribution of input parameters q

Question: How sensitive are results to distribution, which is typically not known?

Partition eigenvalues: $C = W \Lambda W^T$

$$\Lambda = \begin{bmatrix} \Lambda_1 & \\ & \Lambda_2 \end{bmatrix}, \quad W = [W_1 \quad W_2]$$

Rotated Coordinates:

$$y = W_1^T q \in \mathbb{R}^n \quad \text{and} \quad z = W_2^T q \in \mathbb{R}^{p-n}$$

Active Variables

Active Subspace: Range of eigenvectors in W_1

Gradient-Based Active Subspace Construction

Active Subspace: Construction based on random sampling

1. Draw M independent samples $\{q^j\}$ from ρ
2. Evaluate $\nabla_q f_j = \nabla_q f(q^j)$
3. Approximate outer product

   $$C \approx \widetilde{C} = \frac{1}{M}\sum_{j=1}^{M}(\nabla_q f_j)(\nabla_q f_j)^T$$

   Note: $\widetilde{C} = GG^T$ where $G = \frac{1}{\sqrt{M}}[\nabla_q f_1, ..., \nabla_q f_M]$

4. Take SVD of $G = W\sqrt{\Lambda}V^T$
   - Active subspace of dimension n is first n columns of W

One Goal: Develop efficient algorithm for codes that do not have adjoint capabilities

Note: Finite-difference approximations tempting but not effective for high-D

Strategy: Algorithm based on initialized adaptive Morris indices

Gradient-Based Active Subspace Construction

Example: Consider

$$y = e^{c_1 q_1 + c_2 q_2} = f(q)$$

so

$$\nabla_q f(q) = \begin{bmatrix} c_1 e^{c_1 q_1 + c_2 q_2} \\ c_2 e^{c_1 q_1 + c_2 q_2} \end{bmatrix} = \begin{bmatrix} c_1 f(q) \\ c_2 f(q) \end{bmatrix}$$

For $Q_1, Q_2 \sim \mathcal{U}(0,1)$, we have

$$\begin{aligned} C &= \int_0^1 \int_0^1 (\nabla_q f)(\nabla_q f)^T dq_1 dq_2 \\ &= \int_0^1 \int_0^1 \begin{bmatrix} c_1^2 f^2(q) & c_1 c_2 f^2(q) \\ c_1 c_2 f^2(q) & c_2^2 f^2(q) \end{bmatrix} dq \\ &= \begin{bmatrix} c_1^2 & c_1 c_2 \\ c_1 c_2 & c_2^2 \end{bmatrix} \cdot \frac{1}{4 c_1 c_2} \left(e^{2c_1} - 1 \right) \left(e^{2c_2} - 1 \right) \\ &= \begin{bmatrix} \frac{c_1}{4c_2} & \frac{1}{4} \\ \frac{1}{4} & \frac{c_2}{4c_1} \end{bmatrix} \cdot \left(e^{2c_1} - 1 \right) \left(e^{2c_2} - 1 \right) \end{aligned}$$

Values: $c_1 = 0.7$, $c_2 = 0.3$

Analytic C:

$$C = \begin{bmatrix} 1.4652 & 0.6279 \\ 0.6279 & 0.2691 \end{bmatrix}$$

Monte Carlo Approx:

$$C \approx \frac{1}{M} \sum_{j=1}^{M} \left(\nabla_q f(q^j) \right) \left(\nabla_q f(q^j) \right)^T$$

$$M = 10^4$$

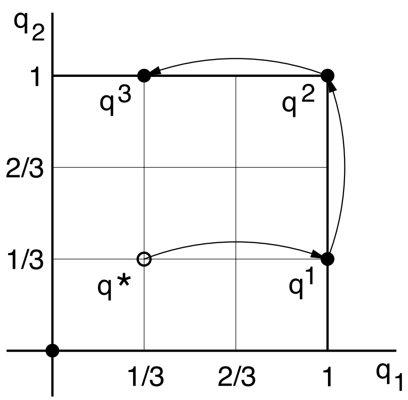
$$C = \begin{bmatrix} 1.4532 & 0.6228 \\ 0.6228 & 0.2669 \end{bmatrix}$$

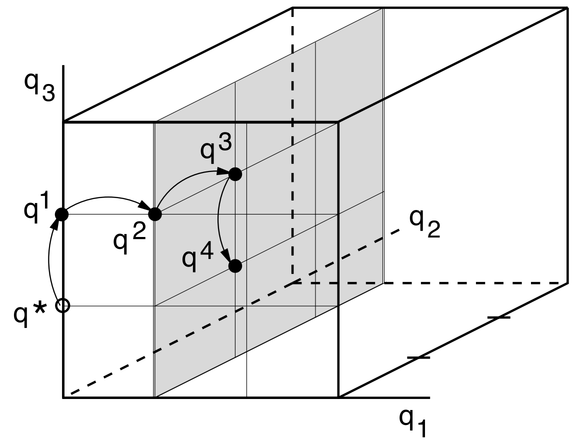
$$M = 10^6$$

$$C = \begin{bmatrix} 1.4654 & 0.6280 \\ 0.6280 & 0.2692 \end{bmatrix}$$

Morris Screening: Random Sampling of Approximated Derivatives

Example: Consider uniformly distributed parameters on $\Gamma = [0,1]^p$





Adaptive Algorithm:

- Use SVD to adapt stepsizes and directions to reflect active subspace.
- Reduce dimension of differencing as active subspace is discovered.

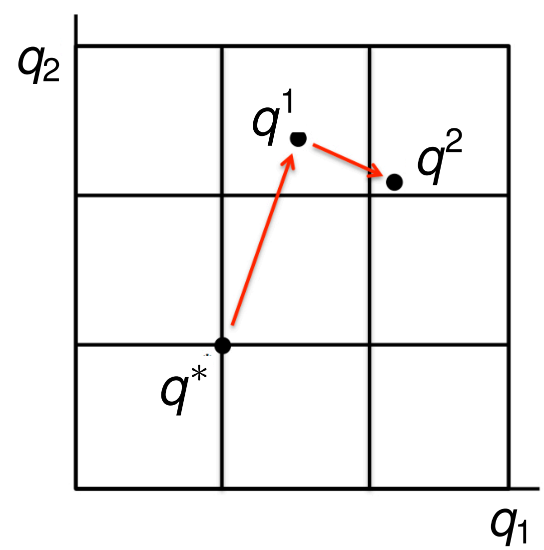
Elementary Effect:

$$d_i = \frac{f(q^j + \Delta e_i) - f(q^j)}{\Delta}$$

Global Sensitivity Measures: r samples

$$\mu_i^* = \frac{1}{r}\sum_{j=1}^{r} |d_i^j(q)|$$

$$\sigma_i^2 = \frac{1}{r-1}\sum_{j=1}^{r}\left(d_i^j(q) - \mu_i\right)^2 \quad , \quad \mu_i = \frac{1}{r}\sum_{j=1}^{r} d_i^j(q)$$



Note: Gets us to moderate-D but initialization required for high-D

Initialization Algorithm

1. Inputs: ℓ iterations, h function evaluations per iteration
2. Sample w^1 from surface of unit sphere where approximately linear

For $j = 1, \dots, \ell$

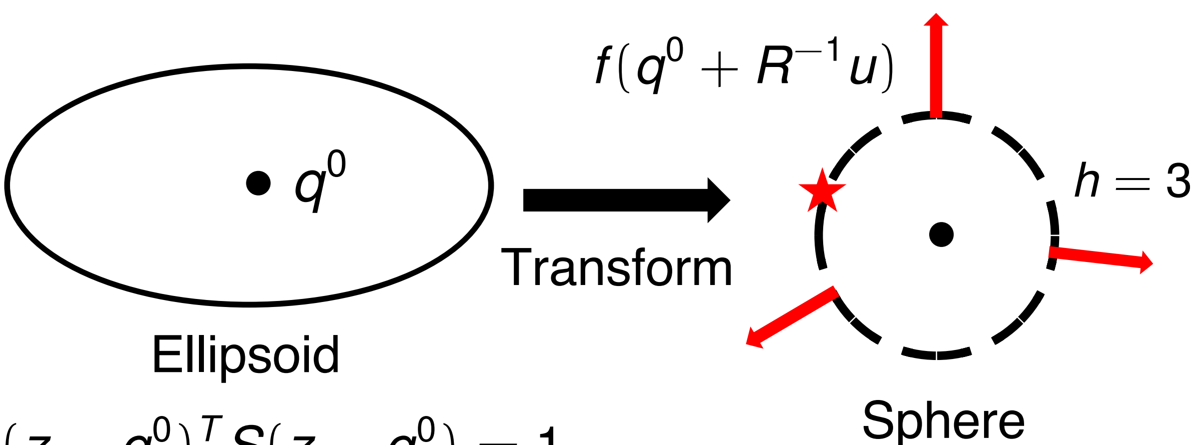
3. Sample $\{\tilde{v}_1^j, \dots, \tilde{v}_h^j\}$ from surface of sphere
4. Use Lagrange multiplier to determine

$$u_{\max}^j = a_0^+ w^j + \sum_{i=1}^{h} a_i^+ v_i^j \ , \ v_i^1 = \tilde{v}_i^1$$

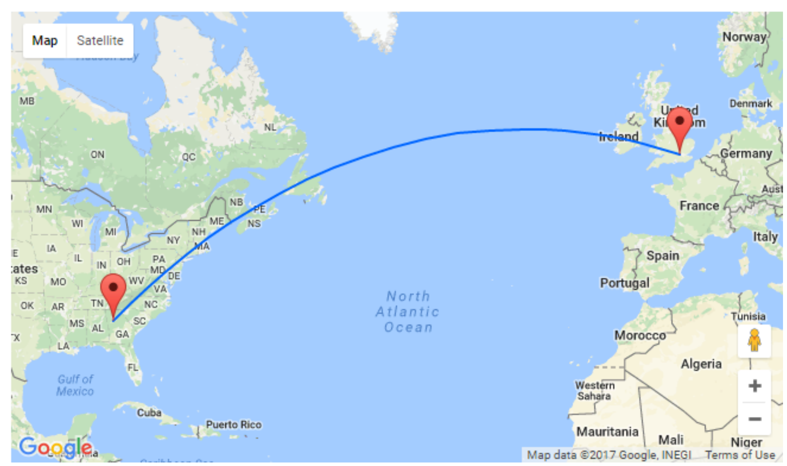
that maximizes $g(u) = f(q^0 + R^{-1}u)$.

Note: For h=1, maximizing great circle through w^1, v^1

Example: Let w^1 = Atlanta, v^1 = London, and g(u) = 'QUIETness' of seatmate on flight



$$(z - q^0)^T S (z - q^0) = 1$$
$$S = R^T R$$



Initialization Algorithm

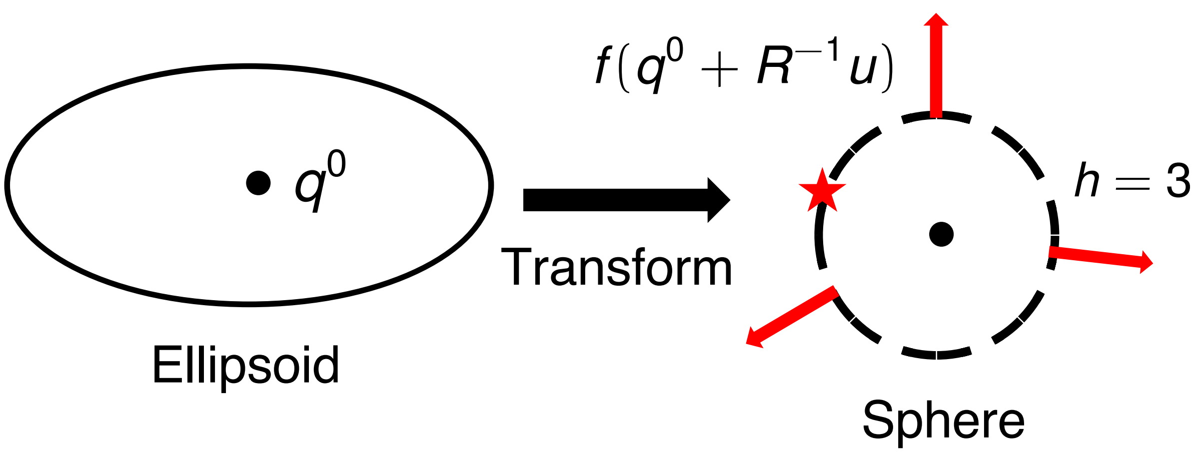
1. Inputs: ℓ iterations, h function evaluations per iteration
2. Sample w^1 from surface of unit sphere where approximately linear

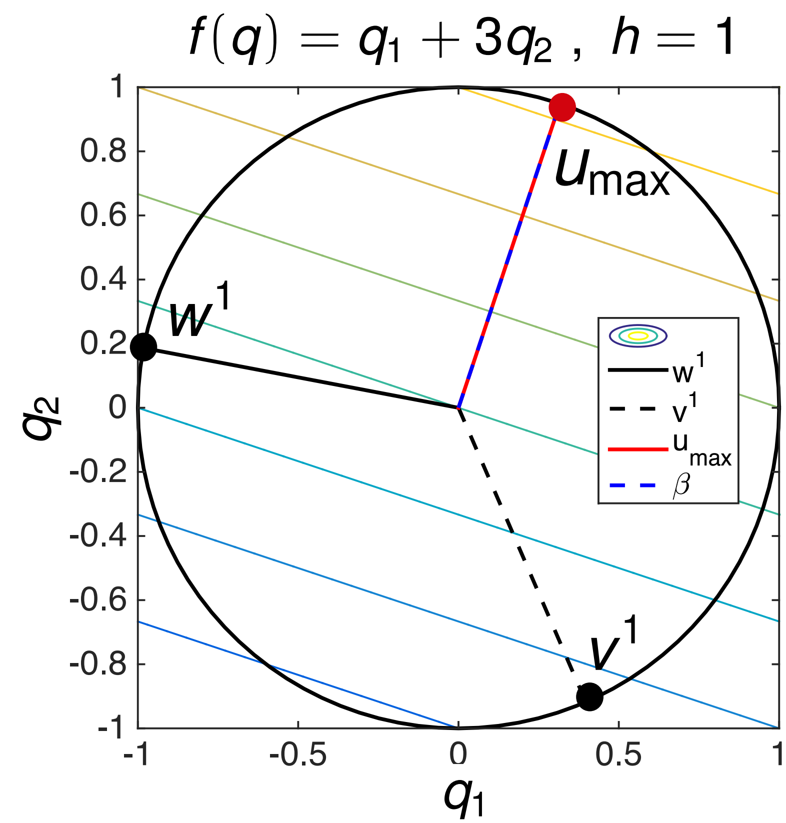
For $j = 1, \dots, \ell$

3. Sample $\{\tilde{v}_1^j, \dots, \tilde{v}_h^j\}$ from surface of sphere
4. Use Lagrange multiplier to determine

$$u_{\max}^j = a_0^+ w^j + \sum_{i=1}^{h} a_i^+ v_i^j \ , \ v_i^1 = \tilde{v}_i^1$$

that maximizes $g(u) = f(q^0 + R^{-1}u)$.





Initialization Algorithm

1. Inputs: ℓ iterations, h function evaluations per iteration
2. Sample w^1 from surface of unit sphere where approximately linear

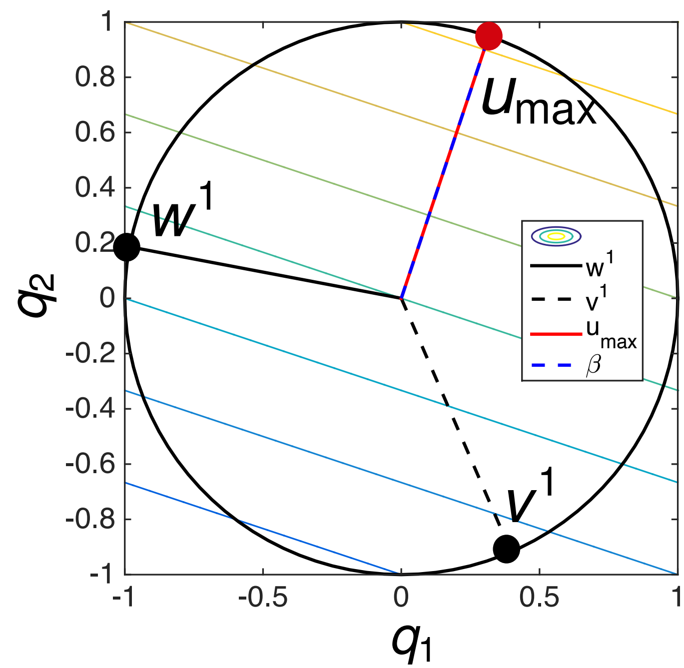
For $j = 1, \ldots, \ell$

3. Sample $\{\tilde{v}_1^j, \ldots, \tilde{v}_h^j\}$ from surface of sphere
4. Use Lagrange multiplier to determine

$$u_{\max}^j = a_0^+ w^j + \sum_{i=1}^{h} a_i^+ v_i^j \ , \ v_i^1 = \tilde{v}_i^1$$

that maximizes $g(u) = f(q^0 + R^{-1}u)$.

Set $w^{j+1} = u_{\max}^j$.



5. Take $C = [w^j, v_1^j, \ldots, v_h^j]$ and set $P_{u_{\max}^j} = u_{\max}^j (u_{\max}^j)^T$
6. Let $C_{j\perp} = \left[\text{span}\left(C_{(j-1)\perp}, (I_m - P_{u_{\max}^j} C\right)\right]$ and set $P_{C_{j\perp}} = C_{j\perp}(C_{j\perp}^T C_{j\perp})^{-1} C_{j\perp}^T$
7. Take $v_i^j = \dfrac{(I_m - P_{C_{j\perp}})\tilde{v}_i^j}{\|(I_m - P_{C_{j\perp}})\tilde{v}_i^j\|}$, $i = 1, \ldots, h$ and repeat

Ortho-complement of $u_{\max}$

Example: Initialization Algorithm to Approximate Gradient

Example: Family of elliptic PDE's

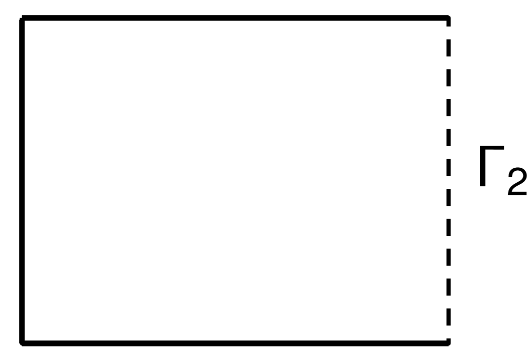
$$-\nabla_s \cdot (a(q,s,\ell)\nabla_s u(s,a(q,s,\ell))) = 1 \ , \ s = [0,1]^2 \ , \ \ell = 1, \cdots, n$$

with the random field representations

$$a(q,s,\ell) = a_{\min} + e^{\bar{a}(s,\ell)+\sum_{i=1}^{p} q_k^\ell \gamma_i \phi_i(s)}$$

Quantity of interest: e.g., strain along edge at n levels

$$f\left(\mathbf{q}^1, ..., \mathbf{q}^n\right) \approx \sum_{\ell=1}^{n} \frac{1}{|\Gamma_2|} \int_{\Gamma_2} u(q,s,\ell)ds$$

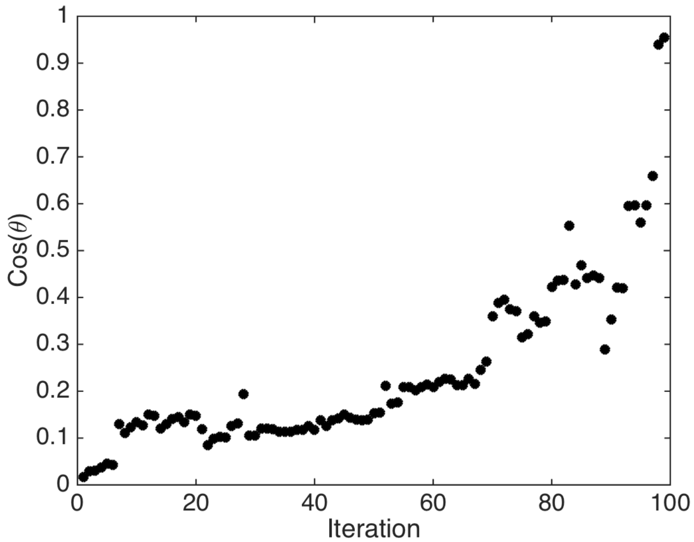


Problem Dimensions:

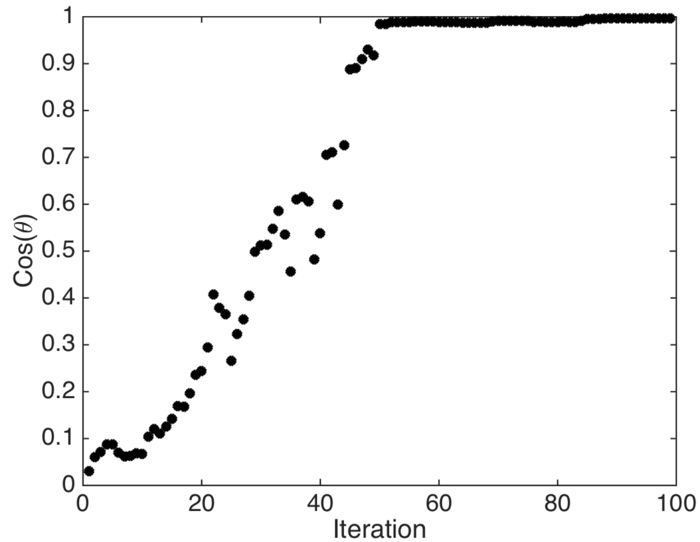
- Parameter dimension: p = 100
- Active subspace dimension: n = 1
- Finite element approximation

Example: Initialization Algorithm to Approximate Gradient

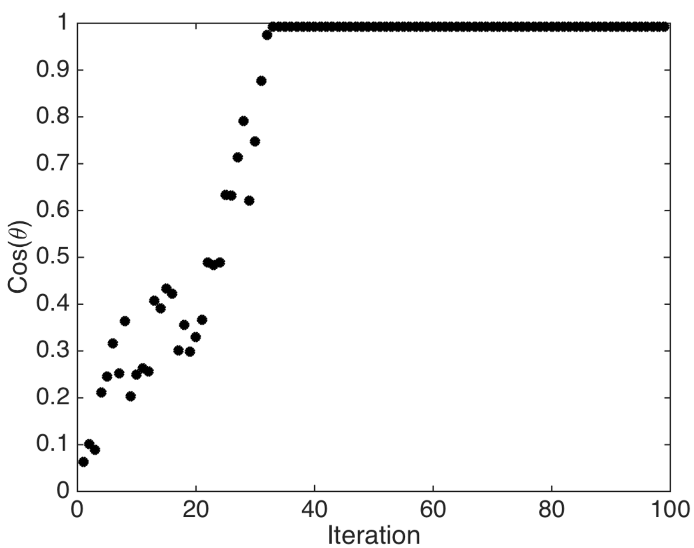
Results: Cosine of angle between ’analytic’ and computed gradient



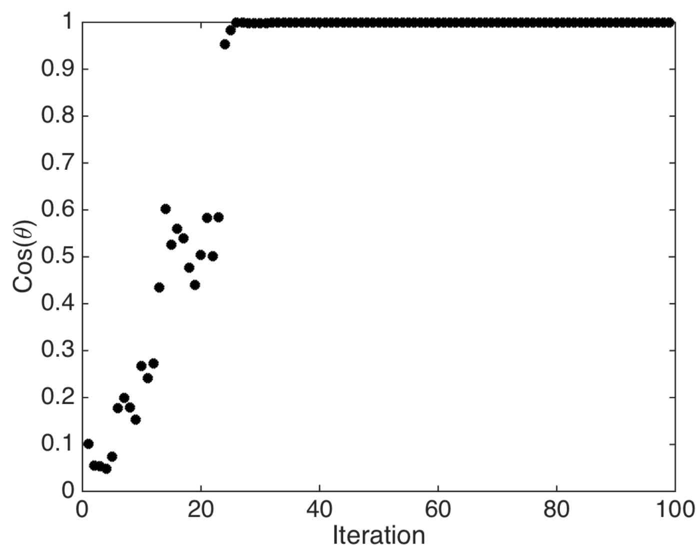
$h = 1$



$h = 2$



$h = 3$



$h = 4$

Recall: p=100

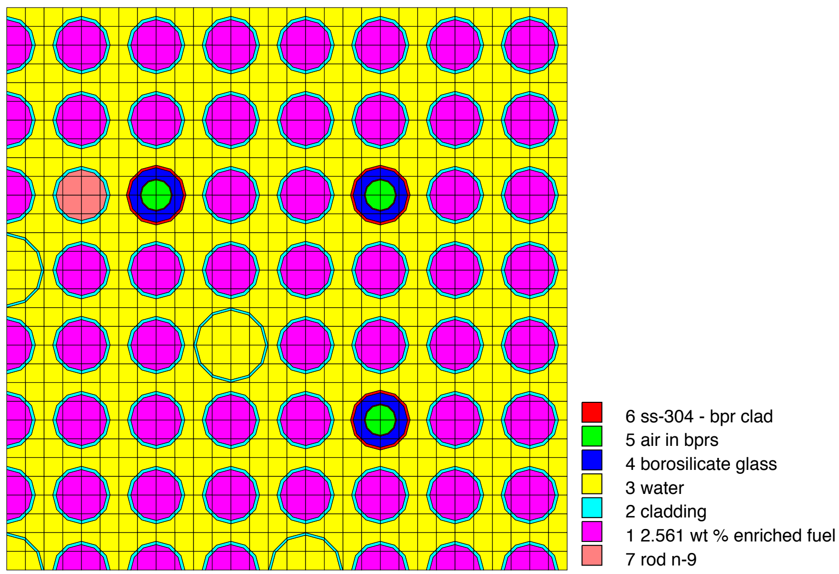
Note: Convergence within $h \cdot \ell$ iterations

SCALE6.1: High-Dimensional Example

Setup: Cross-section computations SCALE6.1

- Input Dimension: 7700
- Output k_{eff}

Materials			Reactions	
$^{234}_{92}\mathrm{U}$	$^{10}_{5}\mathrm{B}$	$^{31}_{15}\mathrm{P}$	Σ_t	$n \to \gamma$
$^{235}_{92}\mathrm{U}$	$^{11}_{5}\mathrm{B}$	$^{55}_{25}\mathrm{Mn}$	Σ_e	$n \to p$
$^{236}_{92}\mathrm{U}$	$^{14}_{7}\mathrm{N}$	$_{26}\mathrm{Fe}$	Σ_f	$n \to d$
$^{238}_{92}\mathrm{U}$	$^{15}_{7}\mathrm{N}$	$^{116}_{50}\mathrm{Sn}$	Σ_c	$n \to t$
$^{1}_{1}\mathrm{H}$	$^{23}_{11}\mathrm{Na}$	$^{120}_{50}\mathrm{Sn}$	$\bar{\nu}$	$n \to {}^3\mathrm{He}$
$^{16}_{8}\mathrm{O}$	$^{27}_{13}\mathrm{Al}$	$_{40}\mathrm{Zr}$	χ	$n \to \alpha$
$_{6}\mathrm{C}$	$_{14}\mathrm{Si}$	$_{19}\mathrm{K}$	$n \to n'$	$n \to 2n$



PWR Quarter Fuel Lattice

Note: We cannot efficiently approximate all directional derivatives required to approximate the gradient matrix. Requires efficient initialization algorithm.

SCALE6.1: High-Dimensional Example

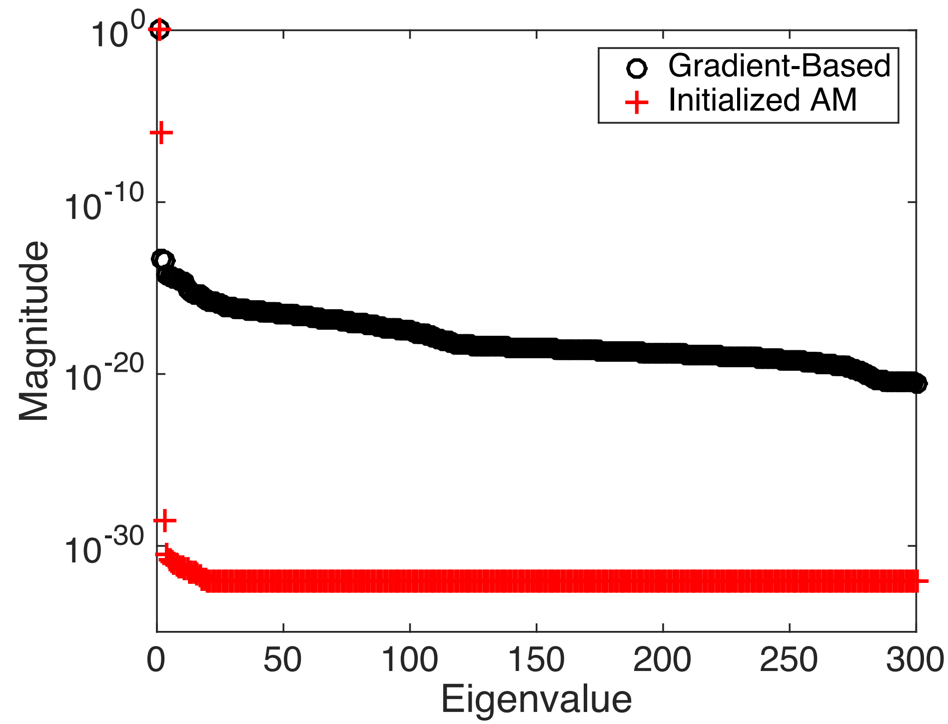
Setup:

- Input Dimension: 7700

SCALE Evaluations:

- Gradient-Based: 1000
- Initialized Adaptive Morris: 18,392
- Projected Finite-Difference: 7,701,000

Note: Analytic eigenvalues: 0, 1



Active Subspace Dimensions:

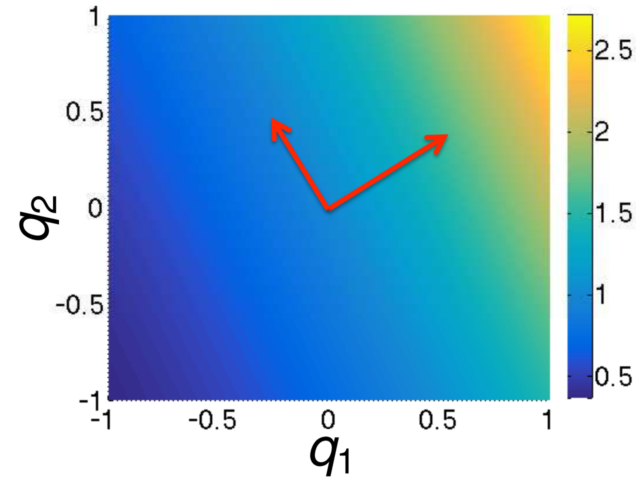
For surrogate sampled off space

	Gap	PCA				Error Tolerance			
Method		0.75	0.90	0.95	0.99	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Gradient-Based	1	2	6	9	24	1	13	90	233
Initialized AM	1	1	1	1	2	1	2	2	2

Notes: Computing *converged* adjoint solution is expensive and *often not achieved*

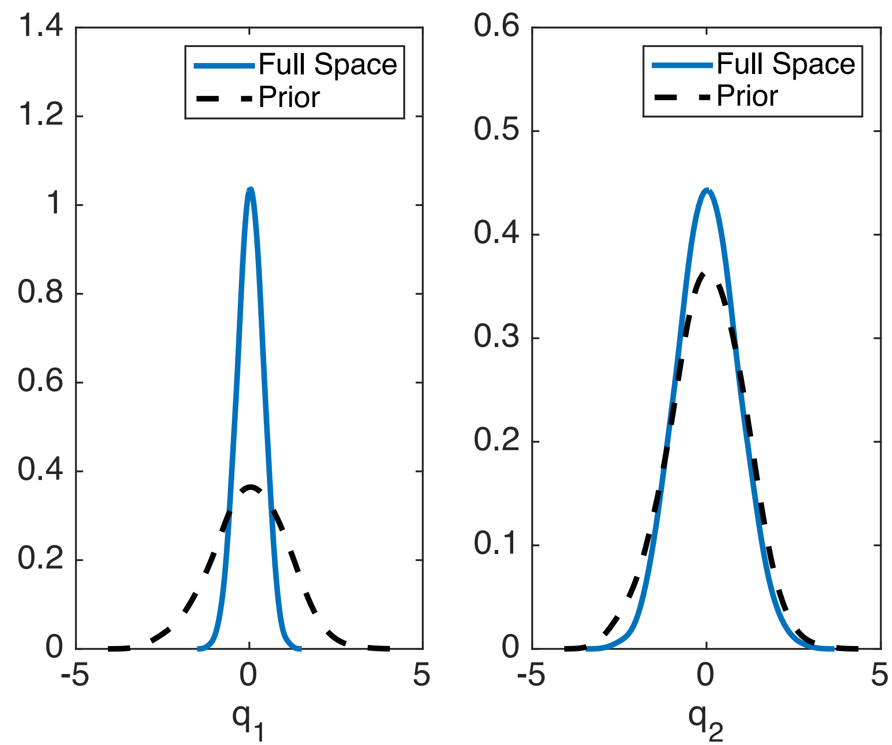
Bayesian Inference on Active Subspaces

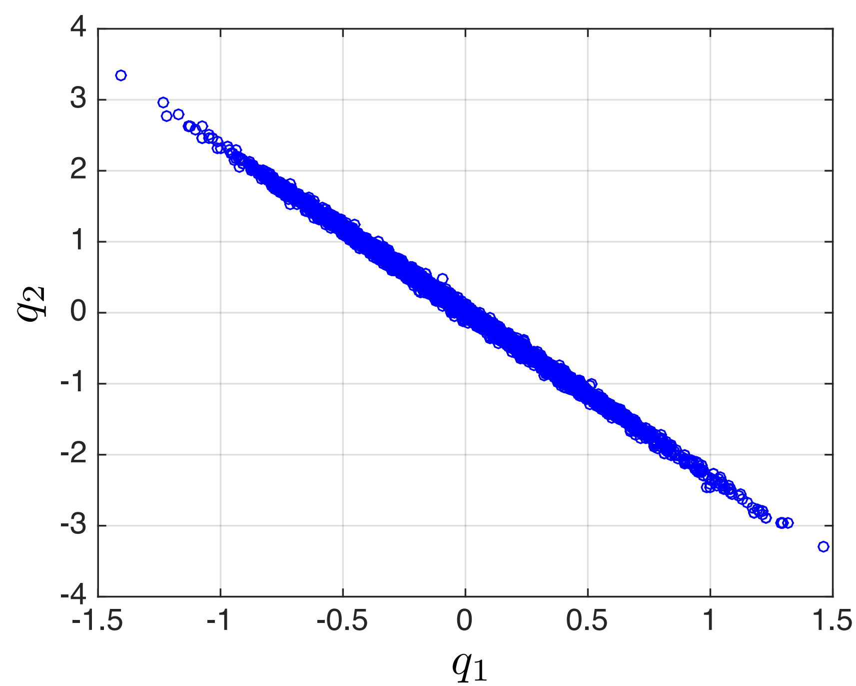
Example: $y = \exp(0.7q_1 + 0.3q_2)$



Full Space Inference:

- Parameters not jointly identifiable
- Result: Prior for 2nd parameter is minimally informed.
- Goal: Use active subspace to quantify parameter sensitivity and guide inference.





Bayesian Inference on Active Subspaces

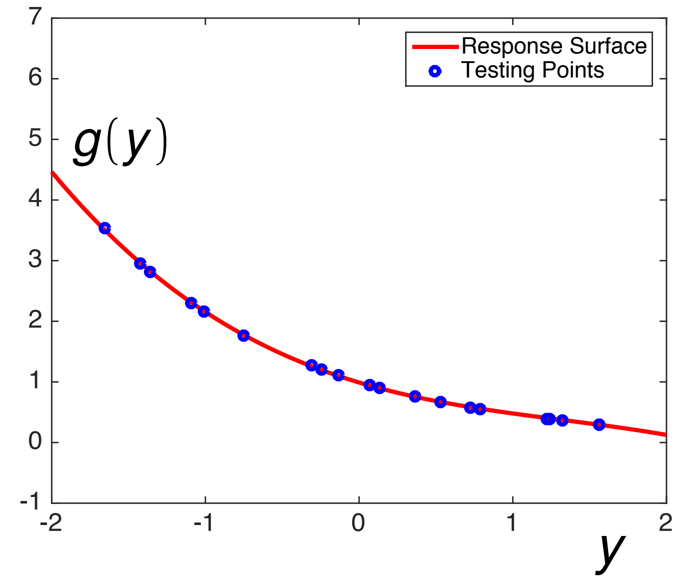
Example: $y = \exp(0.7q_1 + 0.3q_2)$

Active Subspace: For gradient matrix G, form SVD

$$G = U\Lambda V^T$$

Eigenvalue spectrum indicates 1-D active subspace with basis

$$U(:,1) = [0.91\ ,\ 0.39]$$

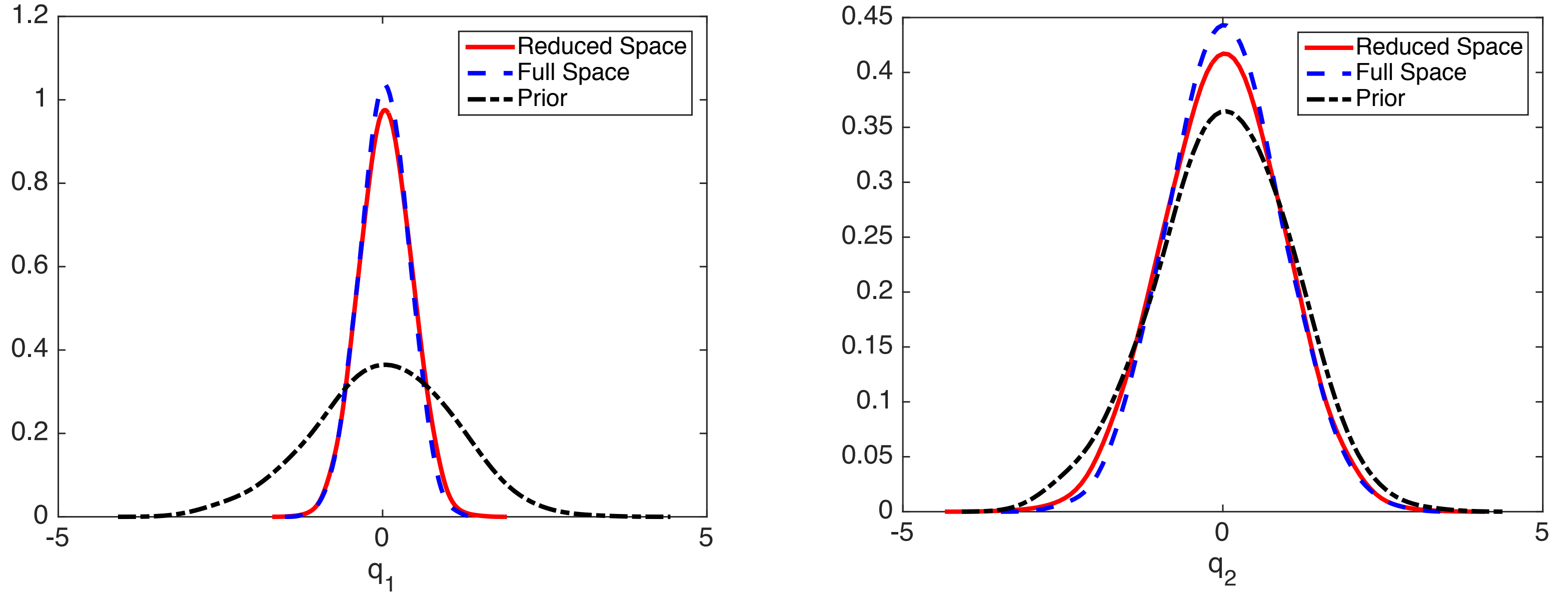


Strategy: Inference based on active subspace

- For values $\{q^j\}_{j=1}^M$, compute $y^j = U(:,1)^T q^j$ and fit response surface $g(y)$
- Use DRAM to calibrate y
- Because model is "invariant" to $z = U(:,2)^T q$, draw $\{z^j\} \sim N(0,1)$
- Transform to $q^j = U(:,1)y^j + U(:,2)z^j$ to obtain posterior densities for physical parameters

Bayesian Inference on Active Subspaces

Results: Inference based on active subspace



Global Sensitivity: For active subspace of dimension N, consider vector of activity scores

$$\alpha(N) = \sum_{j=1}^{N} \lambda_j w_j^2$$

Note: Here N = 1 and $w_j^2 = U(:,1).*U(:,1) = [0.91^2 \ , \ 0.39^2]$

Conclusion: First parameter is more influential and better informed during Bayesian inference.

Bayesian Inference on Active Subspaces

Example: Family of elliptic PDE's

$$-\nabla_s \cdot (a(q,s,\ell)\nabla_s u(s,a(q,s,\ell)) = 1 \ , \ s = [0,1]^2 \ , \ \ell = 1, \cdots, n$$

with the random field representations

$$a(q,s,\ell) = a_{\min} + e^{\overline{a}(s,\ell) + \sum_{i=1}^{p} q_k^{\ell} \gamma_i \phi_i(s)}$$

Quantity of interest: e.g., strain along edge at N levels

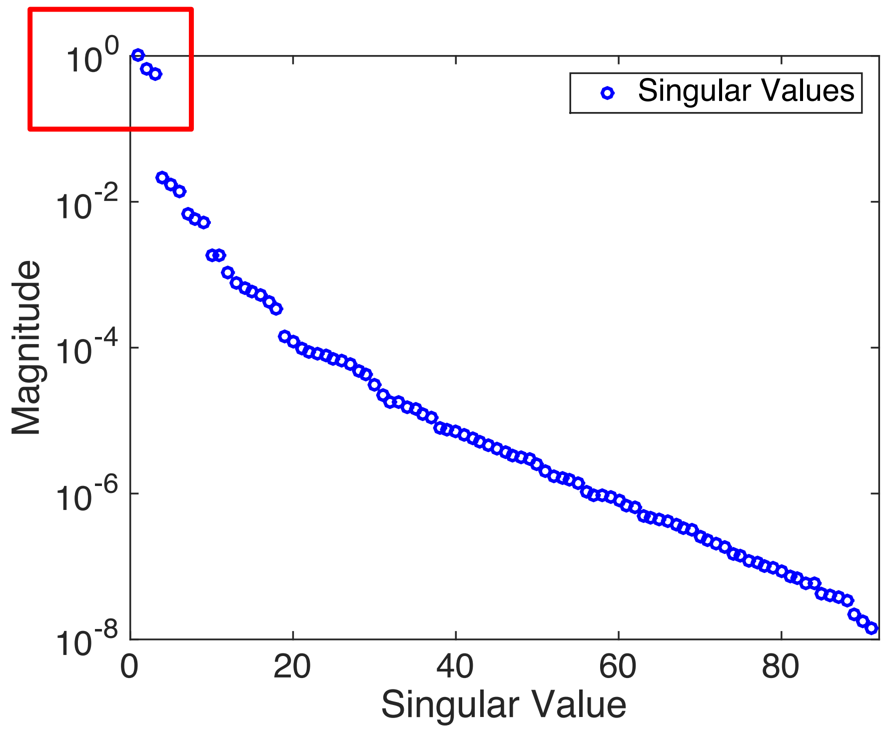
$$f\left(\mathbf{q}^1, \ldots, \mathbf{q}^n\right) \approx \sum_{\ell=1}^{n} \frac{1}{|\Gamma_2|} \int_{\Gamma_2} u(q,s,\ell) ds$$

Problem Dimensions:

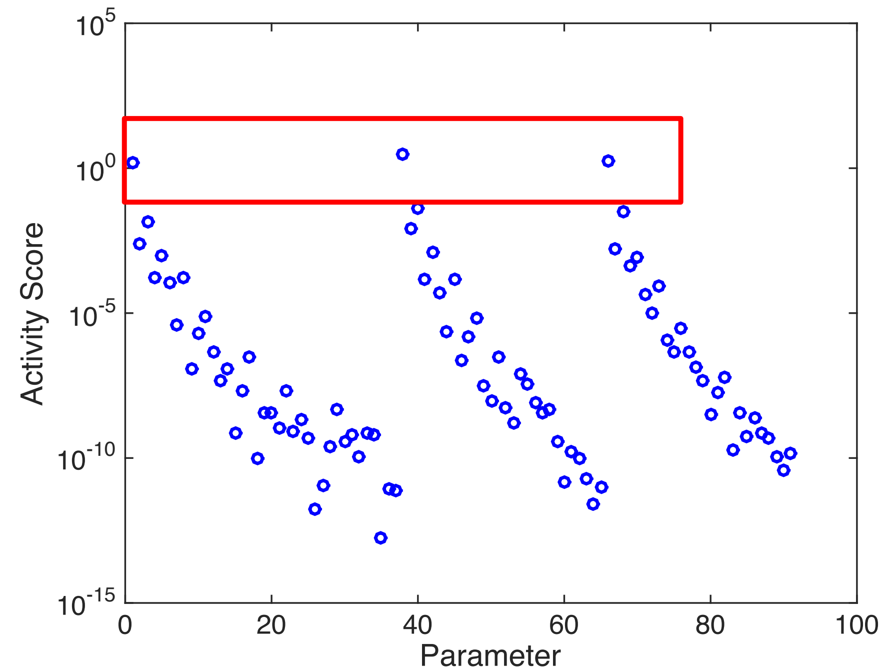
- Parameter dimension: p = 91
- Active subspace dimension: N = 3
- Finite element space: 1372 triangular elements, 727 nodes

Bayesian Inference on Active Subspaces

Singular Values: Recall N = 3



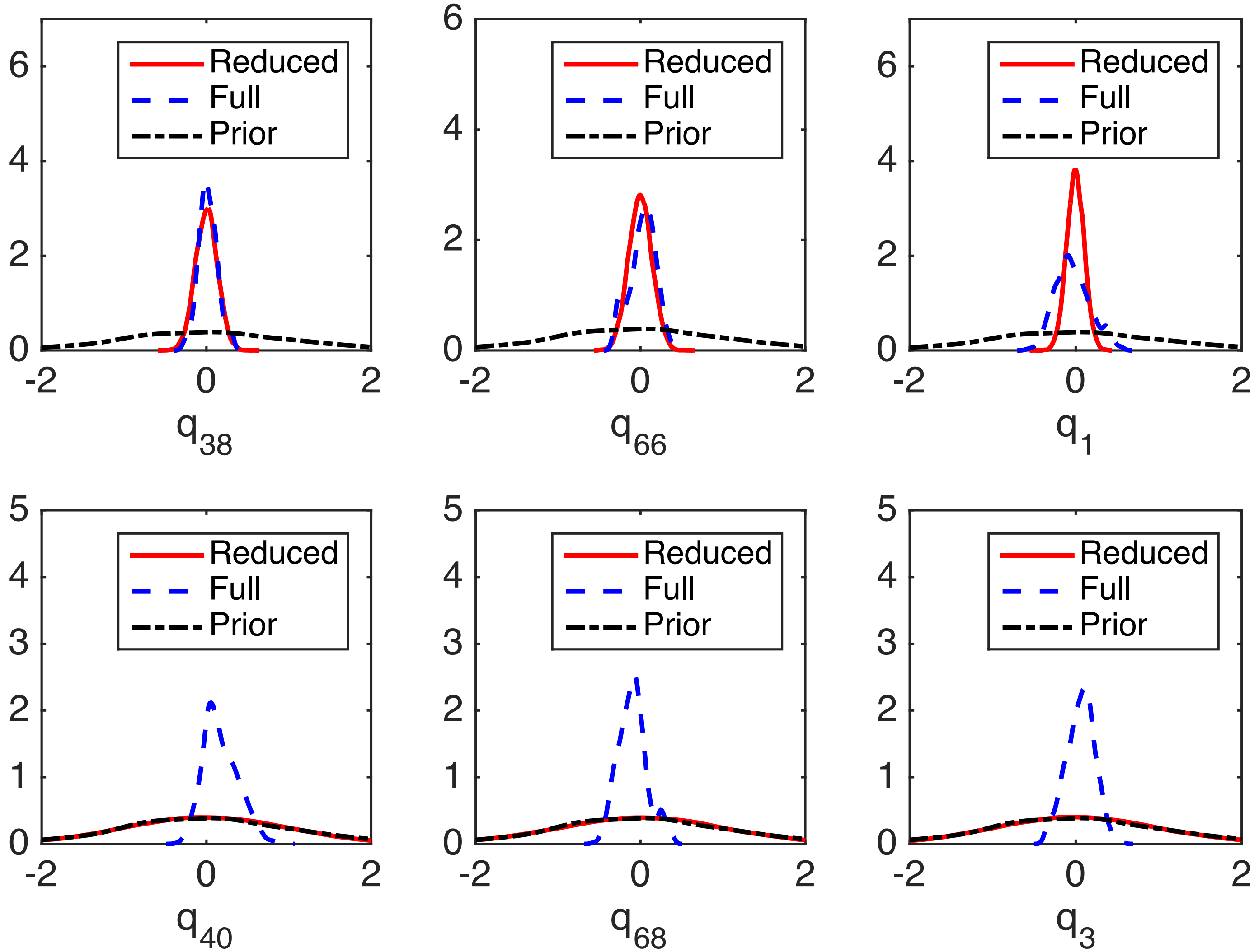
Activity Scores: Quantify global sensitivity



Conclusion: Parameters 1, 38, 66 are most influential and will be primarily informed during Bayesian inference

Bayesian Inference on Active Subspaces

Recall: Parameters 1, 38, 66 are most influential and will be primarily informed during Bayesian inference



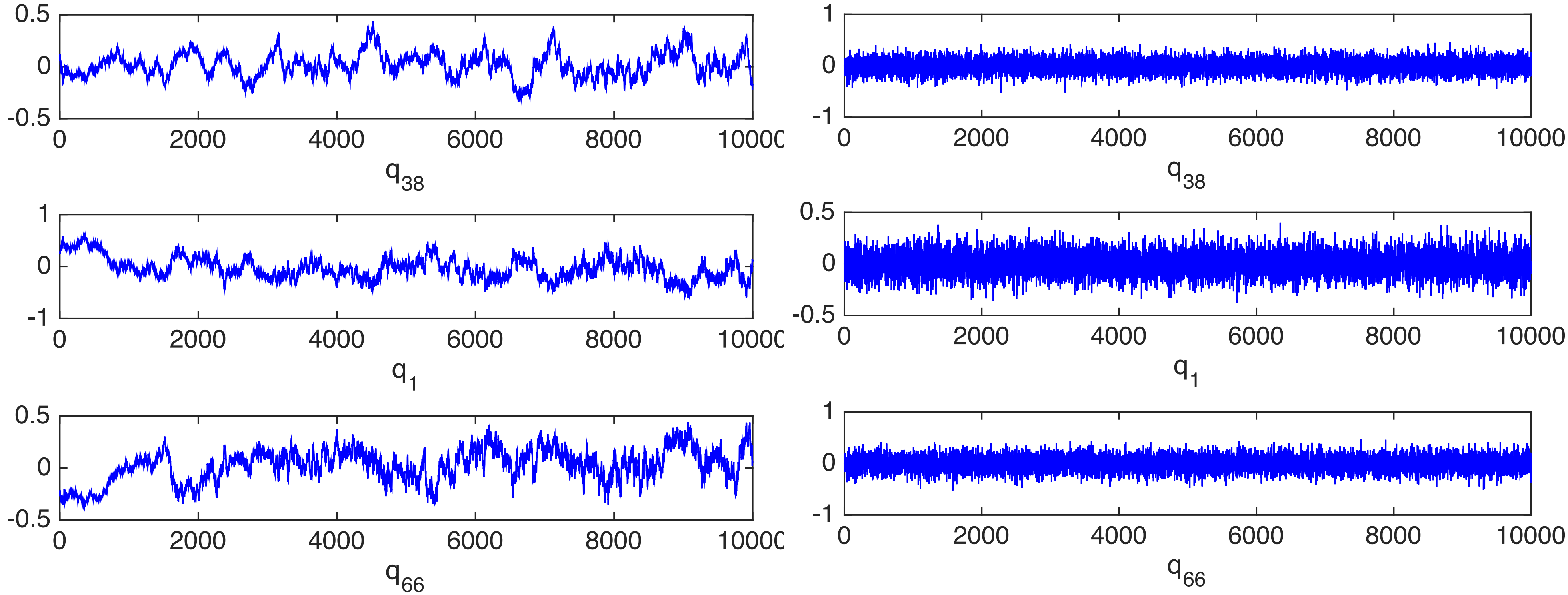
Note:

- Full space: 18 hours
- Reduced: 20 seconds

Bayesian Inference on Active Subspaces

Note:

- Chains for full space not converging well due to parameter nonidentifiability
- Hence full space inference is less reliable



Full Space

Active Subspace