Dimension Reduction in Polynomial Chaos Surrogates using Adaptive DGSM

Manav Vohra[†], Sankaran Mahadevan[†]
Collaborator: Alen Alexanderian[§]

[†]Vanderbilt University
[§]North Carolina State University

December 13, 2017

POLYNOMIAL CHAOS EXPANSION

$$\mathcal{M}(\boldsymbol{x}) \approx \mathcal{M}^{PC}(\boldsymbol{x}) = \sum_{\alpha \in \mathcal{A}} c_\alpha \Psi_\alpha(\boldsymbol{x})$$

Projection-based

$$c_\alpha = \mathbb{E}[\Psi_\alpha(\boldsymbol{x})\mathcal{M}(\boldsymbol{x})]$$

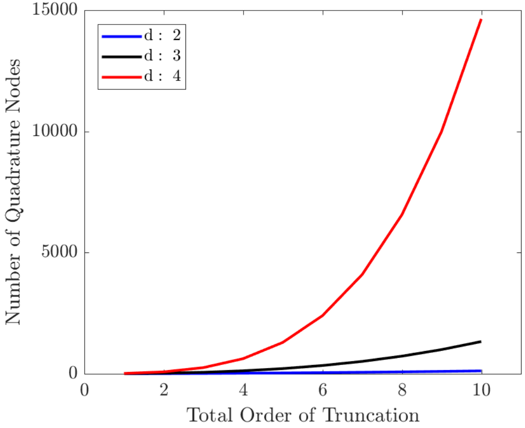
- Gaussian Quadrature
- Smolyak Sparse Quadrature
- Nested Quadrature

Regression-based

$$\hat{\boldsymbol{c}} = \text{argmin}\, \mathbb{E}\left[\left(\boldsymbol{c}^T \Psi(\boldsymbol{x}) - \mathcal{M}(\boldsymbol{x})\right)^2\right]$$

- Ordinary Least-Squares
- Least Angle Regression
- Orthogonal Matching Pursuit

MOTIVATION: DIMENSION REDUCTION



MOTIVATION: DGSM

- Sensitivity analysis based on Sobol indices is commonly used to determine relative importance of the parameters.
- Sobol sensitivity indices are compute intensive:

$$\mathcal{T}(\theta_i) = \frac{\mathbb{E}_{\boldsymbol{\theta}\sim i}[\mathbb{V}_{\theta_i}(\mathcal{G}|\boldsymbol{\theta}_{\sim i})]}{\mathbb{V}(\mathcal{G})}$$

- Bounds on Sobol indices can be computed easily using DGSM and are shown to converge at a much faster rate.

DGSM

- DGSM for Randomly distributed parameters [Sobol and Kucherenko, 2009]:

$$\mu_i = \mathbb{E}\left[\left(\frac{\partial G(\boldsymbol{x})}{\partial x_i}\right)^2\right]$$

where,

$$\frac{\partial G(\boldsymbol{x}^*)}{\partial x_i} = \lim_{\delta \to 0} \frac{[G(x_1^*, \ldots, x_{i-1}^*, x_i^* + \delta, x_{i+1}^*, \ldots, x_d^*) - G(\boldsymbol{x}^*)]}{\delta}$$

- Total number of model realizations required to compute μ_i using N samples is $N(d+1)$.

Bounds on Sobol Indices

- Upper bound (UB_i) on Sobol Total Effect index (ST_i) [Sobol and Kucherenko, 2009]:

$$ST_i \leq \frac{\mathcal{C}_i\mu_i}{V} \; (\propto \hat{\mathcal{C}_i\mu_i})$$

$$\hat{\mathcal{C}_i\mu_i} = \frac{\mathcal{C}_i\mu_i}{\sum_i \mathcal{C}_i\mu_i}$$

$\mathcal{C}$: Poincaré Constant V: Variance

- The Poincaré Constant is specific to a given probability distribution:

$\mathcal{U}[a,b]$	$(b-a)^2/\pi^2$
$\mathcal{N}(\mu,\sigma^2)$	σ^2

REDUCED MORRIS FUNCTION

$$f(x) = \sum_{i=1}^{4} b_i x_i + \sum_{i \leq j}^{4} b_{ij} x_i x_j + \sum_{i \leq j \leq k = 4}^{4} b_{ijk} x_i x_j x_k$$

$$x_i \sim \mathcal{U}[0, 1]$$

$$b_i = \begin{bmatrix} 0.05 \\ 0.59 \\ 10 \\ 0.21 \end{bmatrix}, \; b_{ij} = \begin{bmatrix} 0 & 80 & 60 & 40 \\ 0 & 30 & 0.73 & 0.18 \\ 0 & 0 & 0.64 & 0.93 \\ 0 & 0 & 0 & 0.06 \end{bmatrix}, \; b_{ij4} = \begin{bmatrix} 0 & 10 & 0.98 & 0.19 \\ 0 & 0 & 0.49 & 50 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

PARAMETER SCREENING

ST_i: 10^5 Samples

x_1: 0.4323

x_2: 0.4051

x_4: 0.1300

x_3: 0.0928

PARAMETER SCREENING

UB_i: 5 Samples	ST_i: 10^5 Samples
x_1: 1.4539	x_1: 0.4323
x_2: 1.3303	x_2: 0.4051
x_4: 0.3761	x_4: 0.1300
x_3: 0.3374	x_3: 0.0928

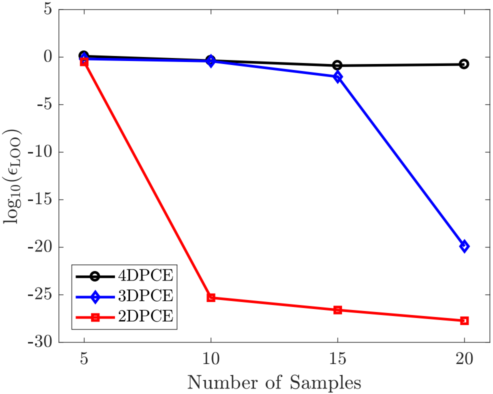
Parameter Screening

UB_i: 5 Samples	UB_i: 10 Samples	ST_i: 10^5 Samples
x_1: 1.4539	x_1: 0.6926	x_1: 0.4323
x_2: 1.3303	x_2: 0.6804	x_2: 0.4051
x_4: 0.3761	x_4: 0.2531	x_4: 0.1300
x_3: 0.3374	x_3: 0.1443	x_3: 0.0928

PCE CONVERGENCE

$$\epsilon_{LOO} = \frac{\sum_{i=1}^{N}\left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \mathcal{M}^{PCE\setminus i}(\boldsymbol{x}^{(i)})\right)^2}{\sum_{i=1}^{N}\left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \hat{\mu}_Y\right)^2}$$

$$\hat{\mu}_Y = \frac{1}{N}\sum_{i=1}^{N}\mathcal{M}(\boldsymbol{x}^{(i)})$$



- The PCE is constructed using Least Angle Regression with Latin Hypercube Sampling.

PCE VERIFICATION

- Generate an independent set of validation points in the 4D input parameter space.
- Compute the following error:

$$\epsilon_{val} = \frac{\left[\sum_{i=1}^{N}\left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \mathcal{M}^{PCE}(\boldsymbol{x}^{(i)})\right)^2\right]^{\frac{1}{2}}}{\left[\sum_{i=1}^{N}\left(\mathcal{M}(\boldsymbol{x}^{(i)})\right)^2\right]^{\frac{1}{2}}}$$

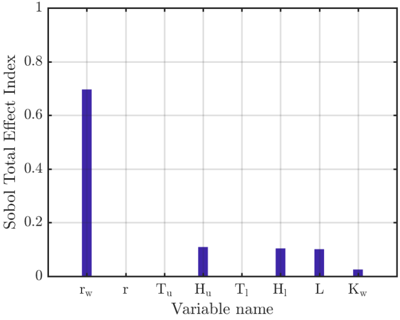
- ϵ_{val} was found to be $\mathcal{O}(10^{-1})$ in the case of 4D PCE and 2D PCE constructed using 15 and 5 training points respectively.

THE BOREHOLE FUNCTION

$$\mathcal{Q} = \frac{2\pi T_u(H_u - H_l)}{\ln\left(\frac{r}{r_w}\right)\left(1 + \frac{2LT_u}{\ln\left(\frac{r}{r_w}\right)r_w^2 K_w} + \frac{T_u}{T_l}\right)}$$

$\mathcal{Q}$: Discharge of water through a borehole

r_w	$\mathcal{N}$(0.1,0.016)	Radius of the borehole (m)
r	$\log\mathcal{N}$(7.71,1.006)	Radius of influence (m)
T_u	$\mathcal{U}$[63070,115600]	Transmissivity of upper aquifer (m^2/yr)
H_u	$\mathcal{U}$[990,1110]	Potentiometric head of upper aquifer (m)
T_l	$\mathcal{U}$[63.1,116]	Transmissivity of lower aquifer (m^2/yr)
H_l	$\mathcal{U}$[700,820]	Potentiometric head of lower aquifer (m)
L	$\mathcal{U}$[1120,1680]	Length of borehole (m)
K_w	$\mathcal{U}$[9855,12045]	Hydraulic conductivity of borehole (m/yr)



Parameter Screening

ST_i: 10^5 Samples

x_1: 0.6942

x_3: 0.1062

x_5: 0.1059

x_6: 0.1026

x_7: 0.0250

x_2: 0.0000

x_4: 0.0000

PARAMETER SCREENING

UB_i: 5 Samples	ST_i: 10^5 Samples
x_1: 0.2124	x_1: 0.6942
x_6: 0.0406	x_3: 0.1062
x_3: 0.0393	x_5: 0.1059
x_5: 0.0393	x_6: 0.1026
x_7: 0.0089	x_7: 0.0250
x_2: 0.0000	x_2: 0.0000
x_4: 0.0000	x_4: 0.0000

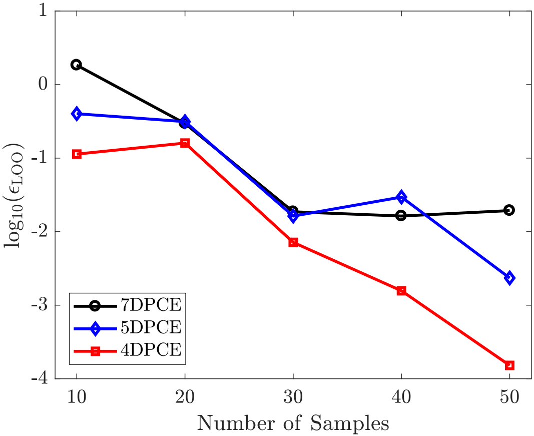
Parameter Screening

UB_i: 5 Samples	UB_i: 10 Samples	ST_i: 10^5 Samples
x_1: 0.2124	x_1: 0.8671	x_1: 0.6942
x_6: 0.0406	x_3: 0.1505	x_3: 0.1062
x_3: 0.0393	x_5: 0.1505	x_5: 0.1059
x_5: 0.0393	x_6: 0.1466	x_6: 0.1026
x_7: 0.0089	x_7: 0.0337	x_7: 0.0250
x_2: 0.0000	x_2: 0.0000	x_2: 0.0000
x_4: 0.0000	x_4: 0.0000	x_4: 0.0000

PARAMETER SCREENING

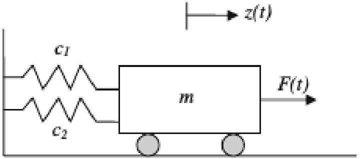
UB_i: 5 Samples	UB_i: 10 Samples	UB_i: 15 Samples	ST_i: 10^5 Samples
x_1: 0.2124	x_1: 0.8671	x_1: 0.8384	x_1: 0.6942
x_6: 0.0406	x_3: 0.1505	x_3: 0.1472	x_3: 0.1062
x_3: 0.0393	x_5: 0.1505	x_5: 0.1472	x_5: 0.1059
x_5: 0.0393	x_6: 0.1466	x_6: 0.1437	x_6: 0.1026
x_7: 0.0089	x_7: 0.0337	x_7: 0.0360	x_7: 0.0250
x_2: 0.0000	x_2: 0.0000	x_2: 0.0000	x_2: 0.0000
x_4: 0.0000	x_4: 0.0000	x_4: 0.0000	x_4: 0.0000

PCE CONVERGENCE AND VERIFICATION



- ϵ_{val} was found to be $\mathcal{O}(10^{-2})$ in the case of 7D PCE and 4D PCE constructed using 60 and 30 training points respectively.

NON-LINEAR OSCILLATOR

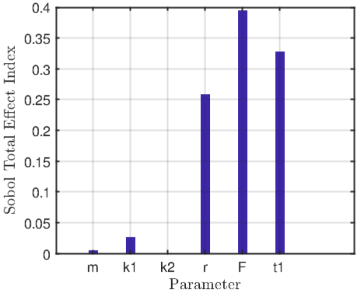


m	$\mathcal{N}$(1,0.05)	Mass
k_1	$\mathcal{N}$(1,0.1)	Spring Constant
k_2	$\mathcal{N}$(0.1,0.01)	Spring Constant
r	$\mathcal{N}$(0.5,0.05)	Displacement
F	$\mathcal{N}$(1,0.2)	Force
t_1	$\mathcal{N}$(1,0.2)	Time

Limit State Function:

$$g(\boldsymbol{X}) = 3r - \left|\frac{2F}{m\omega_0^2}\sin\left(\frac{\omega_0 t_1}{2}\right)\right|$$

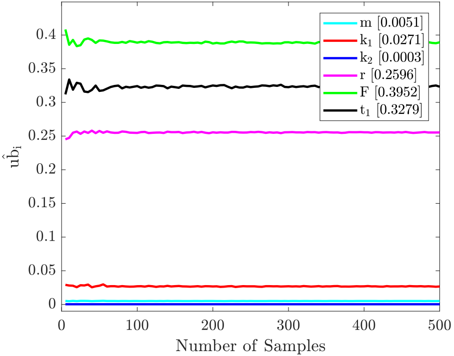
$$\boldsymbol{X} = \{m, k_1, k_2, r, F, t_1\} \qquad \omega_0 = \sqrt{\frac{c_1+c_2}{m}}$$



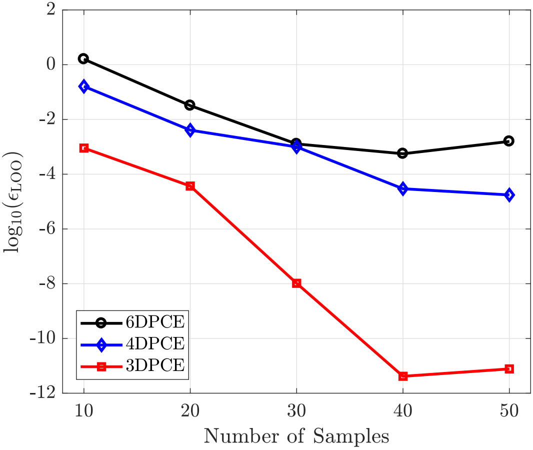
Parameter Rank Metric

$$\widehat{\mathcal{C}_i\mu_i} = \frac{\mathcal{C}_i\mu_i}{\sum_i \mathcal{C}_i\mu_i}$$

$\mathcal{C}$: Poincaré Constant $\quad$ μ_i: DGSM



PCE CONVERGENCE

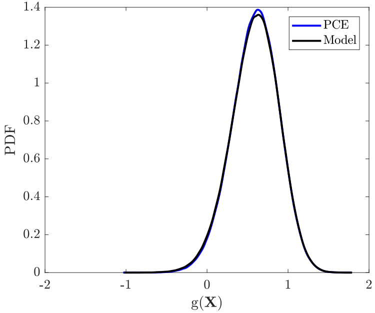


PCE VERIFICATION

- Relative L-2 norm of the error is computed:

$$\epsilon_{val} = \frac{\left[\sum_{i=1}^{N}\left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \mathcal{M}^{PCE}(\boldsymbol{x}^{(i)})\right)^2\right]^{\frac{1}{2}}}{\left[\sum_{i=1}^{N}\left(\mathcal{M}(\boldsymbol{x}^{(i)})\right)^2\right]^{\frac{1}{2}}} = 8.35 \times 10^{-2}$$

- Comparison of PDFs:

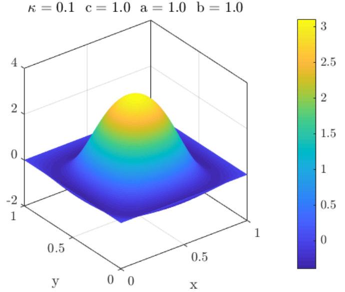


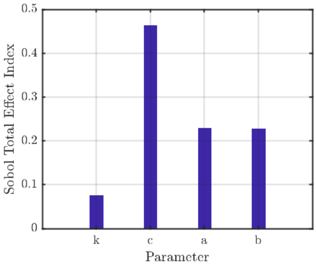
ELLIPTIC PDE

$$-\kappa \Delta u + cu^3 = q$$

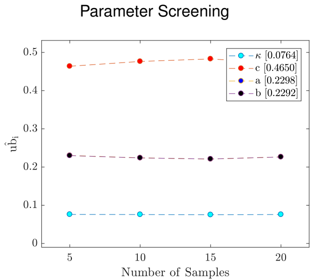
$$q = (-2\pi^2)(a\cos(2\pi x)\sin^2(\pi y) + b\cos(2\pi y)\sin^2(\pi x))$$

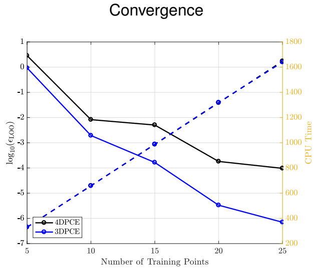
κ	$\mathcal{U}[0.09, 0.11]$
c	$\mathcal{U}[0.9, 1.1]$
a	$\mathcal{U}[0.9, 1.1]$
b	$\mathcal{U}[0.9, 1.1]$

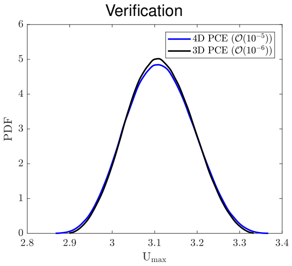




ANALYSIS


Parameter Screening
$\hat{ub}_i$
Number of Samples
κ [0.0764]
c [0.4650]
a [0.2298]
b [0.2292]

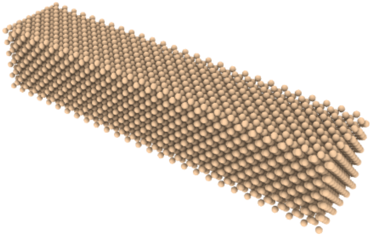

Convergence
$\log_{10}(\epsilon_{LOO})$
CPU Time
Number of Training Points
4DPCE
3DPCE


Verification
PDF
U_{max}
4D PCE ($\mathcal{O}(10^{-5})$)
3D PCE ($\mathcal{O}(10^{-6})$)

ALGORITHM: PARAMETER SCREENING

Generate n_1 points in $\mathbb{R}^d$;
Compute UB_i for parameters θ_i using n_1 points;
% $NF = n_1(d+1)$, NF: Number of model realizations;
Rank Parameters (θ_i) based on UB_i estimates $(\mathcal{R}^{old})$;
set $k = 1$ % k: Iteration counter;
repeat
    Generate βn_1 new points in $\mathbb{R}^d$ $(\beta n_1 \in \mathbb{Z})$;
    Compute UB_i^{new} using $(1+\beta k)n_1$ points;
    % $NF = (1+\beta k)n_1(d+1)$;
    Rank Parameters based on UB_i^{new} estimates $(\mathcal{R}^{new})$;
    if $(\mathcal{R}^{new} = \mathcal{R}^{old})$ **then**
        Compute: $r_i = \frac{UB_i^{new}}{max(UB_i^{new})}$;
        Construct a set $s = \{\theta_i \ni r_i < \alpha\}$;
        Exit the loop;
    end
    set $k = k + 1$;
until $\mathcal{R}^{new} \neq \mathcal{R}^{old}$;
Construct a validation set: $(\boldsymbol{\theta}_j, \mathcal{M}(\boldsymbol{\theta}_j))$, $j=1,2,\ldots,NF$;

APPLICATION: NON-EQUILIBRIUM MD



Lattice Constant (Å)	5.43
W, H (Å)	117.94, 117.94
Temperature (K)	300
Δt (ps)	0.0005
BC	Periodic
Structure	Diamond
Potential, Φ	Stillinger-Weber

$$\Phi = f(A, B, p, q, a, \lambda, \gamma)$$