Uncertainty Analysis for Complex Systems: Algorithms and Challenges

Dongbin Xiu

Department of Mathematics, Purdue University

PURDUE UNIVERSITY

Support: AFOSR, DOE, NNSA, NSF

(Re-)Formulation of PDE: Input Parameterization

$$\frac{\partial u}{\partial t}(t,x) = \mathcal{L}(u) \quad + \text{ boundary/initial conditions}$$

- **Goal:** To characterize the random inputs by a set of random variables
  - Finite number
  - Mutual independence
- **If inputs == parameters**
  - Identify the (smallest) independent set
  - Prescribe probability distribution
- **Else if inputs == fields/processes**
  - Approximate the field by a function of finite number of RVs
  - Well-studied for Gaussian processes
  - Under-developed for non-Gaussian processes
  - Examples: Karhunen-Loeve expansion, spectral decomposition, etc.

$$a(x,\omega) \approx \mu_a(x) + \sum_{i=1}^{d} \tilde{a}_i(x) Z_i(\omega)$$

The Reformulation

- **Stochastic PDE:**

$$\frac{\partial u}{\partial t}(t,x,Z) = \mathcal{L}(u) \quad + \text{ boundary/initial conditions}$$

- **Solution:** $u(t,x,Z):[0,T]\times \bar{D}\times \mathbb{R}^{n_z} \to \mathbb{R}$

- Uncertain inputs are characterized by n_z random variables Z

- Probability distribution of Z is prescribed

$$F_Z(s) = \Pr(Z \le s), \quad s \in \mathbb{R}^{n_z}$$

Non-trivial task

Generalized Polynomial Chaos (gPC)

$$\frac{\partial u}{\partial t}(t,x,Z) = \mathcal{L}(u) \quad + \text{ boundary/initial conditions}$$

- **Focus on dependence on *Z*:** $u(\bullet, Z) : \mathbb{R}^{n_Z} \to \mathbb{R}$

- ***N*th-order gPC expansion:**

$$u_N(t,x,Z) \triangleq \sum_{|\mathbf{k}|=0}^{N} \hat{u}_{\mathbf{k}}(t,x)\Phi_{\mathbf{k}}(Z), \quad \# \text{ of basis} = \begin{pmatrix} n_z + N \\ N \end{pmatrix}$$

- **Orthogonal basis:** $\int \Phi_{\mathbf{i}}(Z)\Phi_{\mathbf{j}}(Z)\rho(Z)\,dZ = \delta_{\mathbf{ij}}$

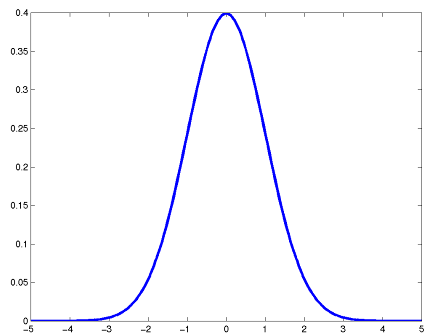
- **Basis functions:**
  - Hermite polynomials: seminal work by *R. Ghanem*
  - General orthogonal polynomials *(Xiu & Karniadakis, 2002)*

- **Properties:**
  - Rigorous mathematics
  - High accuracy, fast convergence
  - Curse-of-dimensionality
- **Numerical Approaches:**
  - Galerkin vs. collocation

gPC Basis

- **Expectation:** $$\mathbb{E}\big(g(Z)\big) = \int_{\mathbb{R}} g(z)\rho(z)\,dz$$

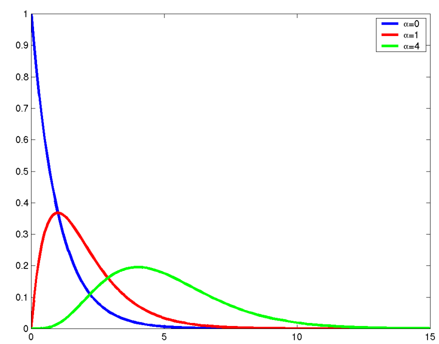
- **Orthogonality:** $$\int \Phi_{\mathrm{i}}(z)\Phi_{\mathrm{j}}(z)\rho(z)\,dz = \mathbb{E}\big[\Phi_{\mathrm{i}}(Z)\Phi_{\mathrm{j}}(Z)\big] = \delta_{\mathrm{ij}}$$



Gaussian distribution

$$\int_{-\infty}^{\infty} \Phi_{\mathrm{i}}(z)\Phi_{\mathrm{j}}(z)e^{-z^2}\,dz = \delta_{\mathrm{ij}}$$

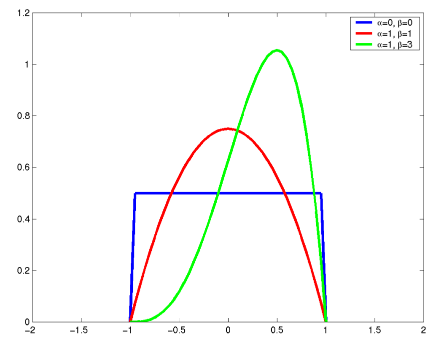
Hermite polynomial



Gamma distribution

$$\int_{0}^{\infty} \Phi_{\mathrm{i}}(z)\Phi_{\mathrm{j}}(z)e^{-z}\,dz = \delta_{\mathrm{ij}}$$

Laguerre polynomial



Beta distribution

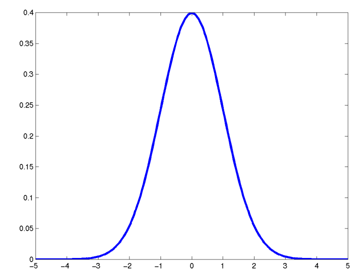
$$\int_{-1}^{1} \Phi_{\mathrm{i}}(z)\Phi_{\mathrm{j}}(z)\,dz = \delta_{\mathrm{ij}}$$

Legendre polynomial

gPC Basis: the Choices

- **Orthogonality:** $$\int \Phi_{\mathbf{i}}(z)\Phi_{\mathbf{j}}(z)\rho(z)\,dz = \mathbb{E}\left[\Phi_{\mathbf{i}}(Z)\Phi_{\mathbf{j}}(Z)\right] = \delta_{\mathbf{ij}}$$

- **Example: Hermite polynomial**



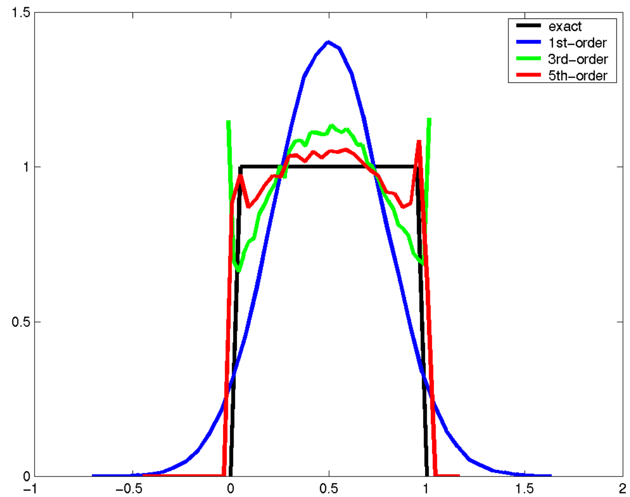
$$\int_{-\infty}^{\infty} \Phi_{\mathbf{i}}(z)\Phi_{\mathbf{j}}(z)e^{-z^2}\,dz = \delta_{\mathbf{ij}}$$

- **The polynomials:** *Z~N(0,1)*

$$\Phi_0 = 1, \quad \Phi_1 = Z, \quad \Phi_2 = Z^2 - 1, \quad \Phi_3 = Z^3 - 3Z, \quad \cdots$$

- **Approximation of arbitrary random variable:** Requires L^2 integrability

- **Example:** Uniform random variable
  - Convergence
  - Non-optimal
  - First-order Legendre is exact



Stochastic Galerkin

$$\frac{\partial u}{\partial t}(t,x,Z) = \mathcal{L}(u) \quad + \text{ boundary/initial conditions}$$

- **Galerkin method:** Seek

$$u_N(t,x,Z) \triangleq \sum_{|\mathbf{k}|=0}^{N} \hat{u}_{\mathbf{k}}(t,x)\Phi_{\mathbf{k}}(Z)$$

Such that

$$\mathbb{E}\left[\frac{\partial u_N}{\partial t}(t,x,Z)\mathbf{\Phi}_{\mathbf{m}}(Z)\right] = \mathbb{E}\left[\mathcal{L}(u_N)\mathbf{\Phi}_{\mathbf{m}}(Z)\right], \quad \forall |\mathbf{m}| \le N$$

- **The result:**
  - Residue is orthogonal to the gPC space
  - A set of deterministic equations for the coefficients
  - The equations are usually coupled – requires new solver

Stochastic Galerkin: An Example

- Equation : $$\frac{du}{dt} = -k(Z)u, \quad u\big|_{t=0} = u_0. \qquad k(Z) = \sum_{i=0}^{N} k_i \Phi_i(Z)$$

  k(Z) is the decaying coefficient with a given probability distribution.

- Seek gPC approximation :

$$v_N(t,Z) = \sum_{i=0}^{N} \hat{v}_i(t)\Phi_i(Z)$$

- Galerkin equation :

$$\frac{d\hat{v}_k}{dt} = -\sum_{i=0}^{N}\sum_{j=0}^{N} e_{ijk} k_i \hat{v}_j, \quad k = 0,\ 1,\ 2,\ldots,N$$

$$e_{ijk} = \int \Phi_i(z)\Phi_j(z)\Phi_k(z)\rho(z)\,dz$$

- Computational complexity: *(N+1)* coupled deterministic ODEs

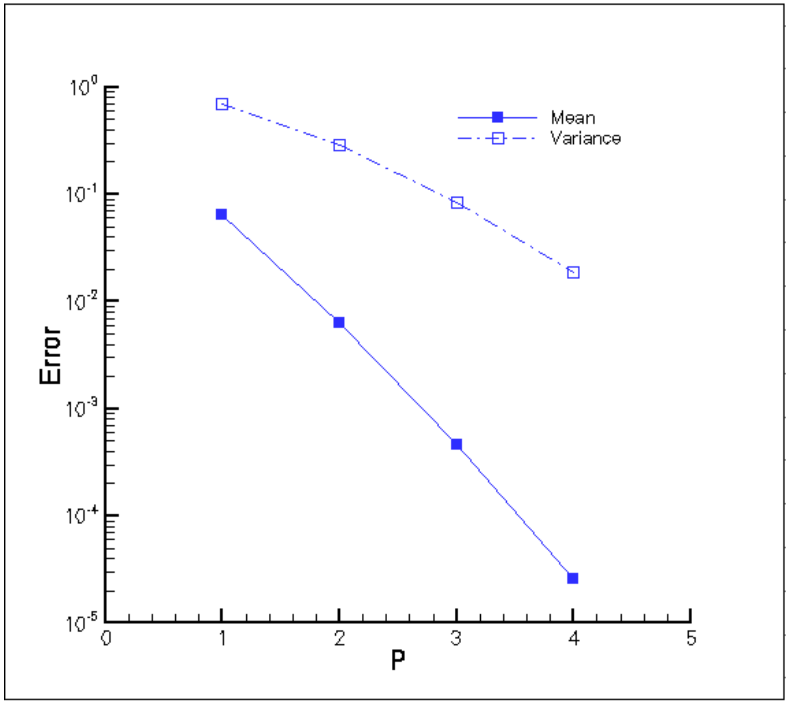
Computational Efficiency

- $du/dt = - k\, u, \;\; u(t=0)=1$

  - k is a **Gaussian** random variable :

    PDF: $$f_k(x) = \frac{1}{\sqrt{2p}} e^{-\frac{x^2}{2}}$$

  - 4^{th}-order **Hermite** expansion



Error	Monte Carlo Method (# of realizations)	Generalized Polynomial Chaos (# of expansion terms)	Speed-up factor
4%	100	1	100
1.1%	1,000	2	500
0.05%	9,800	3	3,267

Stochastic Collocation

$$\frac{\partial u}{\partial t}(t,x,Z) = \mathcal{L}(u) \quad + \text{ boundary/initial conditions}$$

- **Collocation:** To satisfy governing equations at selected nodes
  - Allow one to use existing deterministic codes repetitively

---

- **Sampling:** (solution statistics only)
  - Random (Monte Carlo)
  - Deterministic (lattice rule, tensor grid, cubature)

- **Stochastic collocation:** To construct **polynomial approximations**
  - Node selection is critical to efficiency and accuracy
  - More than sampling

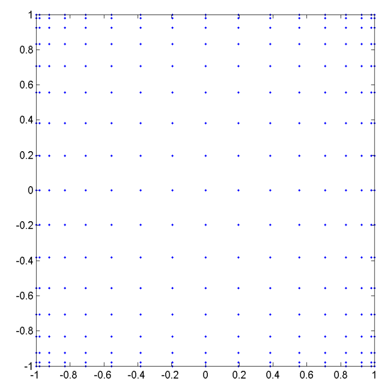
<u>Definition:</u> Given a set of nodes and solution ensemble, find *p(Z)* in a proper polynomial space, such that $p \approx u$ in a proper sense.

Stochastic Collocation: Interpolation

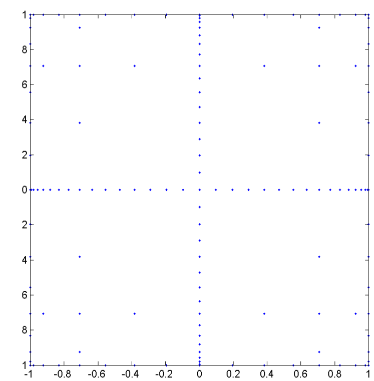
- **Lagrange interpolation:**
  - Let z_j be the nodes and $u(z_j)$ be solution, then Lagrange interpolation

$$p(z) = \sum_{j=1}^{Q} u(z_j) L_j(z) \qquad L_i(z_j) = \delta_{ij}, \quad 1 \le i, j \le N_p$$

- Difficult for unstructured grids.
- Dimension-by-dimension space filling



Tensor grids: inefficient



Sparse grids: more efficient

- **Matrix inversion:** $p(Z) = \sum_{i=1}^{M} c_m \Phi_m(Z)$

$$p(z_j) = \sum_{i=1}^{M} c_m \Phi_m(z_j) = f_j \quad \Rightarrow \quad \mathbf{Ac} = \mathbf{f}$$

Vandermonde matrix: $\mathbf{A} = \left(a_{jk}\right) = \left(\Phi_k(z_j)\right), \quad j = 1, \dots, N_p, \quad k = 1, \dots, M$

Stochastic Collocation: Non-interpolating

- **Regression type:**

$$\min \|\mathbf{Ac} - \mathbf{f}\|$$

  - Over-determined system: least-square type
  - Under-determined system: l_1-minimization, compressive sampling, etc.

- **Discrete projection:**

$$\mathbb{P}_N u = \sum_{|\mathbf{k}|=0}^{N} \hat{u}_{\mathbf{k}}(t,x)\Phi_{\mathbf{k}}(Z)$$

$$\hat{u}_{\mathbf{k}} = \mathbb{E}[u(Z)\Phi_{\mathbf{k}}(Z)] = \int u(z)\Phi_{\mathbf{k}}(z)\rho(z)\,dz$$

$$\approx \sum_{j=1}^{N_p} u(z_j)\Phi_{\mathbf{k}}(z_j)w_j$$

Stochastic Computation: The Landscape

- **Realistic Large-scale Complex Systems:**
  - Complex physics ➔ highly nonlinear systems
  - Large number of random variables
  - (Extremely) time consuming simulations
  - Legacy codes (nearly impossible to re-write)

- **Stochastic Galerkin:**
  - Difficult to implement
  - Good mathematical properties

- **Stochastic collocation is more proper:**
  - Easy to implement ➔ virtually no coding effort
  - Nonlinearity poses no additional difficulties

  - **Easy implementation:**
    1. Choose a set of nodes, Z_j, $j=1,\ldots,N_p$.
    2. Run deterministic simulation at each node Z_j.
    3. Construct polynomial approximation (surrogate/response surface).

Stochastic Computation: Challenges

- **Curse-of-Dimensionality:**
  - Number of simulations grows (too) fast with dimensionality
  - Current approaches:
    - Adaptive (sparse) grid
    - "Sparser" grids
  - Significantly "delayed" but far from satisfactory
    - A rather extreme (but not uncommon) scenario:

      "What if I have 30 random inputs but can only afford 10 simulations?"

- **Do we know all the probability distributions?**
  - In many practical systems, we do not ➔ Epistemic uncertainty
  - Very few studies
  - (Probably) the first numerical approach: *Jakeman, et al, JCP 2010*

- **Multi-physics, multi-scale systems**

Stochastic Computation: "Useful" Algorithms

- **"Useful" UQ algorithms need to target ….**
  - ***Realistic Large-scale Complex Systems:***
    - *Complex physics ➔ highly nonlinear systems*
    - *Large number of random variables*
    - *(Extremely) time consuming simulations*
    - *Legacy codes (nearly impossible to re-write)*

- **More development of "capability-based" UQ**
  - To make UQ algorithms with certain capability/accuracy more efficient
  - For example: adaptive refinement

- **In need of "capacity-based" UQ**
  - To design the "best" method for a given simulation capacity
  - For example:
    - *"What if I have 30 random inputs but can only afford 10 simulations?"*
    - Rephrase: *"Assume we can afford 10 simulations, what can we achieve?"*

Summary

- Uncertainty Analysis: To provide improved prediction
  - Input characterization
  - Uncertainty propagation
  - Post processing
- Generalized polynomial chaos (gPC)
  - Multivariate approximation theory
- Active directions:
  - Compressive sampling
  - Adaptive algorithms
  - Model-form uncertainty
  - Utilization of data: data assimilation, inference, etc.
  - etc, etc, etc…
- **What about visualization?**
  - **Lack of dialogue between the UQ and Viz communities**