Fast Numerical Methods for Stochastic Computations

A review by Dongbin Xiu

Alejandro Pozo

May 16^{th}, 2013

Outline

Example: Burgers' Equation

Let us consider the Burger's equation:

$$\begin{cases} u_t + uu_x = \nu u_{xx}, \quad x \in [-1, 1] \\ u(-1) = 1 \\ u(1) = -1 \end{cases}$$

Example: Burgers' Equation

Let us consider the Burger's equation:

$$\begin{cases} u_t + uu_x = \nu u_{xx}, & x \in [-1, 1] \\ u(-1) = 1 \\ u(1) = -1 \end{cases}$$

It has an exact steady-state solution:

$$u(x) = -A \tanh\left[\frac{A}{2\nu}(x - z)\right]$$

Example: Burgers' Equation

Let us consider the Burger's equation:

$$\begin{cases} u_t + uu_x = \nu u_{xx}, & x \in [-1,1] \\ u(-1) = 1 + \delta \\ u(1) = -1 \end{cases}$$

It has an exact steady-state solution:

$$u(x) = -A \tanh\left[\frac{A}{2\nu}(x - z)\right]$$

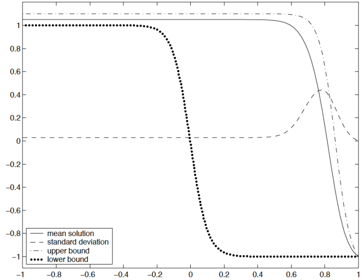
Example: Burgers' Equation

Let us consider the Burger's equation:

$$\begin{cases} u_t + uu_x = \nu u_{xx}, & x \in [-1,1] \\ u(-1) = 1 + \delta \\ u(1) = -1 \end{cases}$$

It has an exact steady-state solution:

$$u(x) = -A \tanh\left[\frac{A}{2\nu}(x - z)\right]$$



Techniques

1. **Monte Carlo and sampling methods**

   Generate independent realizations of random inputs based on the prescribed PDF and extract statistical information.

   - ✓ Straightforward to apply.
   - × Large number of executions needed.

Techniques

1. **Monte Carlo and sampling methods**

   Generate independent realizations of random inputs based on the prescribed PDF and extract statistical information.

   - ✓ Straightforward to apply.
   - × Large number of executions needed.

2. **Perturbation methods**

   Expand (Taylor) random fields around their mean and truncate at a given order.

   - × Small number of uncertainties. Complicated systems of equations beyond 2nd order.

Techniques

1. **Monte Carlo and sampling methods**

   Generate independent realizations of random inputs based on the prescribed PDF and extract statistical information.

   - ✓ Straightforward to apply.
   - × Large number of executions needed.

2. **Perturbation methods**

   Expand (Taylor) random fields around their mean and truncate at a given order.

   - × Small number of uncertainties. Complicated systems of equations beyond 2nd order.

3. **Moment equations**

   Compute moments of the random solution directly from the averages of the original governing equations.

   - × Closure problem: Higher moments are needed for the derivation of a moment.

Techniques

1. **Monte Carlo and sampling methods**

   Generate independent realizations of random inputs based on the prescribed PDF and extract statistical information.
   - ✓ Straightforward to apply.
   - × Large number of executions needed.

2. **Perturbation methods**

   Expand (Taylor) random fields around their mean and truncate at a given order.
   - × Small number of uncertainties. Complicated systems of equations beyond 2nd order.

3. **Moment equations**

   Compute moments of the random solution directly from the averages of the original governing equations.
   - × Closure problem: Higher moments are needed for the derivation of a moment.

4. **Generalized polynomial chaos (gPC)**

   Express stochastic solutions as orthogonal polynomials of the input random parameters.
   - ✓ Fast convergence when the solution depends smoothly on the random parameters.

Techniques

1. **Monte Carlo and sampling methods**

   Generate independent realizations of random inputs based on the prescribed PDF and extract statistical information.
   - ✓ Straightforward to apply.
   - × Large number of executions needed.

2. **Perturbation methods**

   Expand (Taylor) random fields around their mean and truncate at a given order.
   - × Small number of uncertainties. Complicated systems of equations beyond 2nd order.

3. **Moment equations**

   Compute moments of the random solution directly from the averages of the original governing equations.
   - × Closure problem: Higher moments are needed for the derivation of a moment.

4. **Generalized polynomial chaos (gPC)**

   Express stochastic solutions as orthogonal polynomials of the input random parameters.
   - ✓ Fast convergence when the solution depends smoothly on the random parameters.

5. **Operator based methods**

   Manipulate the stochastic operators in the governing equations (Neumann expansion, weighted integral method...)
   - × Small uncertainties. Dependent on the operator. Limited to static problems.

Example: Burgers' Equation (II)

Let us consider the Burger's equation:

$$\begin{cases} u_t + uu_x = 0.05u_{xx}, & x \in [-1,1] \\ u(-1) = 1 + \delta, & \delta \sim \mathcal{U}(0, 0.1) \\ u(1) = -1 \end{cases}$$

Example: Burgers' Equation (II)

Let us consider the Burger's equation:

$$\begin{cases} u_t + uu_x = 0.05u_{xx}, & x \in [-1,1] \\ u(-1) = 1+\delta, & \delta \sim \mathcal{U}(0,0.1) \\ u(1) = -1 \end{cases}$$

Monte Carlo method with n realizations vs. gPC fourth-order expansions:

	$n = 100$	$n = 1000$	$n = 2000$	$n = 5000$	$n = 10000$	gPC
$\bar{z}$	0.819	0.814	0.815	0.814	0.814	0.814
σ_z	0.387	0.418	0.417	0.417	0.414	0.414

Example: Burgers' Equation (II)

Let us consider the Burger's equation:

$$\begin{cases} u_t + uu_x = 0.05u_{xx}, & x \in [-1, 1] \\ u(-1) = 1 + \delta, & \delta \sim \mathcal{U}(0, 0.1) \\ u(1) = -1 \end{cases}$$

Monte Carlo method with n realizations vs. gPC fourth-order expansions:

	$n = 100$	$n = 1000$	$n = 2000$	$n = 5000$	$n = 10000$	gPC
$\bar{z}$	0.819	0.814	0.815	0.814	0.814	0.814
σ_z	0.387	0.418	0.417	0.417	0.414	0.414

Perturbation method of order k vs. gPC fourth-order expansions:

	$k = 1$	$k = 2$	$k = 3$	$k = 4$	gPC
$\bar{z}$	0.823	0.824	0.824	0.824	0.814
σ_z	0.349	0.349	0.328	0.328	0.414

Example: Burgers' Equation (II)

Let us consider the Burger's equation:

$$\begin{cases} u_t + uu_x = 0.05u_{xx}, & x \in [-1,1] \\ u(-1) = 1 + \delta, & \delta \sim \mathcal{U}(0, 0.1) \\ u(1) = -1 \end{cases}$$

Monte Carlo method with n realizations vs. gPC fourth-order expansions:

	$n = 100$	$n = 1000$	$n = 2000$	$n = 5000$	$n = 10000$	gPC
$\bar{z}$	0.819	0.814	0.815	0.814	0.814	0.814
σ_z	0.387	0.418	0.417	0.417	0.414	0.414

Perturbation method of order k vs. gPC fourth-order expansions:

	$k = 1$	$k = 2$	$k = 3$	$k = 4$	gPC
$\bar{z}$	0.823	0.824	0.824	0.824	0.814
σ_z	0.349	0.349	0.328	0.328	0.414

Monte Carlo needs much more computations to obtain same accuracy as gPC (gPC needs the equivalent to five deterministic simulations). Perturbation methods do not even seem to converge.

Governing equations and probabilistic framework

Let us consider:

$$\begin{cases} L(x, u; y) = 0, & \text{in } D, \\ B(x, u; y) = 0, & \text{on } \partial D, \end{cases}$$

where:

- L is a differential operator
- B is a boundary operator (Dirichlet, Neumann...)
- $x = (x_1, \ldots, x_d) \in D \subset \mathbb{R}^d$ are the spatial coordinates
- $y = (y_1, \ldots, y_N) \in \mathbb{R}^N$ are the parameters of interest —random and mutually independent—, defined in $(\Omega, \mathcal{A}, \mathcal{P})$. They can be physical parameters of the system, continuous random processes on the boundary, random initial conditions...

Governing equations and probabilistic framework

Let us consider:

$$\begin{cases} L(x,u;y) = 0, & \text{in } D, \\ B(x,u;y) = 0, & \text{on } \partial D, \end{cases}$$

where:

- L is a differential operator
- B is a boundary operator (Dirichlet, Neumann...)
- $x = (x_1, \dots, x_d) \in D \subset \mathbb{R}^d$ are the spatial coordinates
- $y = (y_1, \dots, y_N) \in \mathbb{R}^N$ are the parameters of interest —random and mutually independent—, defined in $(\Omega, \mathcal{A}, \mathcal{P})$. They can be physical parameters of the system, continuous random processes on the boundary, random initial conditions...

We are interested in a set of quantities (QoI), called observables:

$$g = (g_1, \dots, g_K) \in \mathbb{R}^K = G(u)$$

Governing equations and probabilistic framework

Let us consider:

$$\begin{cases} L(x, u; y) = 0, & \text{in } D, \\ B(x, u; y) = 0, & \text{on } \partial D, \end{cases}$$

where:

- L is a differential operator
- B is a boundary operator (Dirichlet, Neumann...)
- $x = (x_1, \ldots, x_d) \in D \subset \mathbb{R}^d$ are the spatial coordinates
- $y = (y_1, \ldots, y_N) \in \mathbb{R}^N$ are the parameters of interest —random and mutually independent—, defined in $(\Omega, \mathcal{A}, \mathcal{P})$. They can be physical parameters of the system, continuous random processes on the boundary, random initial conditions...

We are interested in a set of quantities (QoI), called observables:

$$g = (g_1, \ldots, g_K) \in \mathbb{R}^K = G(u)$$

Let $\rho_i : \Gamma_i \longrightarrow \mathbb{R}^+$ be the probability density function (PDF) of y_i and

$$\rho(y) = \prod_{i=1}^{N} \rho_i(y_i),$$

the joint PDF of y, with support $\Gamma = \prod_{i=1}^{N} \Gamma_i$.

gPC basis and approximations (I)

One-dimensional orthogonal polynomial spaces in Γ_i:

$$W^{i,d_i} := \{v : \Gamma_i \longrightarrow \mathbb{R} \mid v \in \text{span}\{\phi_m(y_i)\}_{m=0}^{d_i}\}, \quad i = 1, \dots, N$$

where

$$\int_{\Gamma_i} \rho_i(y_i)\phi_m(y_i)\phi_n(y_i)dy_i = h_m^2 \delta_{mn}$$

and

$$h_m^2 = \int_{\Gamma_i} \rho_i \phi_m^2 dy_i$$

gPC basis and approximations (I)

One-dimensional orthogonal polynomial spaces in Γ_i:

$$W^{i,d_i} := \{v : \Gamma_i \longrightarrow \mathbb{R} \mid v \in \mathrm{span}\{\phi_m(y_i)\}_{m=0}^{d_i}\}, \quad i = 1, \dots, N$$

where

$$\int_{\Gamma_i} \rho_i(y_i)\phi_m(y_i)\phi_n(y_i)dy_i = h_m^2 \delta_{mn}$$

and

$$h_m^2 = \int_{\Gamma_i} \rho_i \phi_m^2 dy_i$$

N-dimensional orthogonal polynomial space in Γ:

$$W_N^P := \bigotimes_{|\mathbf{d}| \leq P} W^{i,d_i}$$

where $\mathbf{d} = (d_1, \dots, d_N) \in \mathbb{N}_0^N$ and $|\mathbf{d}| = d_1 + \dots + d_N$. The orthonormal polynomials are constructed as:

$$\Phi_m = \phi_{m_1}(y_1) \dots \phi_{m_N}(y_N), \quad m_1 + \dots + m_N \leq P$$

gPC basis and approximations (II)

Examples:

	Distribution	gPC basis polynomials	Support
Continuous	Gaussian	Hermite	$(-\infty, \infty)$
	Gamma	Laguerre	$[0, \infty)$
	Beta	Jacobi	$[a, b]$
	Uniform	Legendre	$[a, b]$
Discrete	Poisson	Charlier	$\{0, 1, 2, \dots\}$
	Binomial	Krawtchouk	$\{0, 1, \dots, N\}$
	Negative Binomial	Meixner	$\{0, 1, 2, \dots\}$
	Hypergeometric	Hahn	$\{0, 1, \dots, N\}$

gPC basis and approximations (II)

Examples:

	Distribution	gPC basis polynomials	Support
Continuous	Gaussian	Hermite	$(-\infty, \infty)$
	Gamma	Laguerre	$[0, \infty)$
	Beta	Jacobi	$[a, b]$
	Uniform	Legendre	$[a, b]$
Discrete	Poisson	Charlier	$\{0, 1, 2, \dots\}$
	Binomial	Krawtchouk	$\{0, 1, \dots, N\}$
	Negative Binomial	Meixner	$\{0, 1, 2, \dots\}$
	Hypergeometric	Hahn	$\{0, 1, \dots, N\}$

The P^{th}-order gPC approximation of u is:

$$u_N^P(x, y) = \sum_{m=1}^{M} \hat{u}_m(x)\Phi_m(y), \quad M = \binom{N+P}{N}$$

where

$$\hat{u}_m(x) = \int u(x, y)\Phi_m(y)\rho(y)dy = \mathbb{E}[u(x, y)\Phi_m(y)], \quad 1 \leq m \leq M$$

Statistical information

We can compute, for instance, the following statistical information:

- Mean:

$$\mathbb{E}[u](x) \approx \mathbb{E}[u_N^P] = \int \left(\sum_{m=1}^{M} \hat{u}_m(x) \Phi_m(y) \right) \rho(y) dy = \hat{u}_1(x)$$

- Covariance:

$$\begin{aligned} \text{Cov}[u](x_1, x_2) &\approx \mathbb{E}\left[\left(u_N^P(x_1, y) - \mathbb{E}[u_N^P(x_1, y)] \right) \left(u_N^P(x_2, y) - \mathbb{E}[u_N^P(x_2, y)] \right) \right] \\ &= \sum_{m=2}^{M} \hat{u}_m(x_1) \hat{u}_m(x_2) \end{aligned}$$

- Variance:

$$\text{Var}[u](x) \approx \mathbb{E}\left[\left(u_N^P(x, y) - \mathbb{E}[u_N^P(x, y)] \right)^2 \right] = \sum_{m=2}^{M} \hat{u}_m(x)^2$$

- Sensitivity coefficients:

$$\mathbb{E}\left[\frac{\partial u}{\partial y_j} \right] \approx \sum_{m=1}^{M} \left(\hat{u}_m(x) \int \frac{\partial \Phi_m(y)}{\partial y_j} \rho(y) dy \right), \quad j = 1, \ldots, N$$

Galerkin method

We approximate u_N^P by

$$v_N^P(x, y) = \sum_{m=1}^{M} \hat{v}_m(x)\Phi_m(y)$$

such that

$$\int L(x, v_N^P; y)w(y)\rho(y)dy = 0, \quad \text{in } D,$$

$$\int B(x, v_N^P; y)w(y)\rho(y)dy = 0, \quad \text{on } \partial D,$$

for all $w \in W_N^P$.

Galerkin method

We approximate u_N^P by

$$v_N^P(x,y) = \sum_{m=1}^{M} \hat{v}_m(x)\Phi_m(y)$$

such that

$$\int L(x, v_N^P; y) w(y)\rho(y) dy = 0, \quad \text{in } D,$$

$$\int B(x, v_N^P; y) w(y)\rho(y) dy = 0, \quad \text{on } \partial D,$$

for all $w \in W_N^P$.

The resulting equations are a coupled system of M deterministic PDEs for $\{\hat{v}_m\}$.

Collocation methods

- **Lagrange interpolation approach:** Let $\Theta_N = \{y^{(i)}\}_{i=1}^{Q} \in \Gamma$ a set of nodes. Then:

$$u(x,y) \sim \mathcal{I}u(x,y) = \sum_{k=1}^{Q} \tilde{u}_k(x) L_k(y), \quad \forall x \in D$$

where

$$L_i(y^{(j)}) = \delta_{ij} \quad \text{and} \quad \tilde{u}_k(x) = u(x, y^{(k)}), \quad 1 \leq i,j,k \leq Q$$

Collocation methods

- **Lagrange interpolation approach:** Let $\Theta_N = \{y^{(i)}\}_{i=1}^{Q} \in \Gamma$ a set of nodes. Then:

$$u(x,y) \sim \mathcal{I}u(x,y) = \sum_{k=1}^{Q} \tilde{u}_k(x) L_k(y), \quad \forall x \in D$$

where

$$L_i(y^{(j)}) = \delta_{ij} \quad \text{and} \quad \tilde{u}_k(x) = u(x, y^{(k)}), \quad 1 \leq i,j,k \leq Q$$

- **Pseudo-spectral approach:** Let $\bar{\Theta}_N = \{y^{(i)}, \alpha^{(j)}\}_{i=1}^{Q} \in \Gamma$ a set of nodes and weights. Then:

$$w_N^P(x,y) = \sum_{m=1}^{M} \hat{w}_m(x)\Phi_m(y), \quad \text{with} \quad \hat{w}_m(x) = \sum_{j=1}^{Q} u(x, y^{(j)})\Phi_m(y^{(j)})\alpha^{(j)}$$

Collocation methods

- **Lagrange interpolation approach:** Let $\Theta_N = \{y^{(i)}\}_{i=1}^{Q} \in \Gamma$ a set of nodes. Then:

$$u(x,y) \sim \mathcal{I}u(x,y) = \sum_{k=1}^{Q} \tilde{u}_k(x) L_k(y), \quad \forall x \in D$$

where

$$L_i(y^{(j)}) = \delta_{ij} \quad \text{and} \quad \tilde{u}_k(x) = u(x, y^{(k)}), \quad 1 \leq i,j,k \leq Q$$

- **Pseudo-spectral approach:** Let $\bar{\Theta}_N = \{y^{(i)}, \alpha^{(j)}\}_{i=1}^{Q} \in \Gamma$ a set of nodes and weights. Then:

$$w_N^P(x,y) = \sum_{m=1}^{M} \hat{w}_m(x)\Phi_m(y), \quad \text{with} \quad \hat{w}_m(x) = \sum_{j=1}^{Q} u(x, y^{(j)})\Phi_m(y^{(j)}))\alpha^{(j)}$$

In both cases, for each $y^{(k)}$, we have to solve Q uncoupled problems:

$$L(x, \tilde{u}_k; y^{(k)}) = 0, \quad \text{in } D,$$

$$B(x, \tilde{u}_k; y^{(k)}) = 0, \quad \text{on } \partial D,$$

Points selection

It is straightforward in one-dimensional ($N = 1$) problems, where the Gauss quadratures are usually the optimal choice. But, for large ($N \gg 1$) dimensions?

- Tensor products of one-dimensional nodes
- Sparse grids, subsets of the full tensor product based on Smolyak algorithm
- Cubature rules

Thanks for your attention!