A Multilevel Stochastic Collocation Method

Peter Jantsch

University of Tennessee, Knoxville

March 8, 2014

Joint work with Aretha Teckentrup (FSU), Clayton Webster (ORNL/UT), and Max Gunzburger (FSU)

DOE Advanced Simulation Computing Research (ASCR) — #ERKJ259

Model Problem - Linear Elliptic SPDE

Find $u \in L^2_\rho(\Gamma, H^1_0(D))$ such that for almost every $\mathbf{y} \in \Gamma$

$$\nabla \cdot (a(\mathbf{y}, x) \cdot \nabla u(\mathbf{y}, x)) = f(\mathbf{y}, x) \tag{1}$$

We assume that a, f are such that this problem has a unique solution represented in terms of $\mathbf{y} \in \Gamma$, a finite dimensional random vector.

Model Problem - Linear Elliptic SPDE

Find $u \in L^2_\rho(\Gamma, H^1_0(D))$ such that for almost every $\mathbf{y} \in \Gamma$

$$\nabla \cdot (a(\mathbf{y}, x) \cdot \nabla u(\mathbf{y}, x)) = f(\mathbf{y}, x) \tag{1}$$

We assume that a, f are such that this problem has a unique solution represented in terms of $\mathbf{y} \in \Gamma$, a finite dimensional random vector.

Such a PDE might represent ground water flow, etc.

Common Single Level Methods

Monte Carlo Method

- Most popular method
- Simple to implement, easily parallelizable
- Convergence rate $\mathcal{O}(M^{-1/2})$ is dimension independant, but relatively slow

Spectral Galerkin Methods

- Higher rate of convergence
- Degrees of freedom are coupled, leading to a large linear system
- Suffers from the curse of dimensionality

Stochastic Collocation

For stochastic collocation we choose a set of (interpolatory) points $\{\mathbf{y}^j\}_{j=1}^M \subset \Gamma$, and for each $\mathbf{y}^j$ solve the deterministic PDE

$$\nabla \cdot (a(\mathbf{y}^j, x) \cdot \nabla u(\mathbf{y}^j, x)) = f(\mathbf{y}^j, x), \tag{2}$$

using the finite element method to obtain a solution $u_h(\mathbf{y}^j, x)$.

Stochastic Collocation

For stochastic collocation we choose a set of (interpolatory) points $\{\mathbf{y}^j\}_{j=1}^M \subset \Gamma$, and for each $\mathbf{y}^j$ solve the deterministic PDE

$$\nabla \cdot (a(\mathbf{y}^j, x) \cdot \nabla u(\mathbf{y}^j, x)) = f(\mathbf{y}^j, x), \tag{2}$$

using the finite element method to obtain a solution $u_h(\mathbf{y}^j, x)$. Finally, we construct our approximation by Lagrange interpolation:

$$\mathcal{I}_M u_h(\mathbf{y}, x) = \sum_{j=1}^{M} u_h(\mathbf{y}^j, x)\Psi_j(\mathbf{y}) \tag{3}$$

Stochastic Collocation

For stochastic collocation we choose a set of (interpolatory) points $\{\mathbf{y}^j\}_{j=1}^M \subset \Gamma$, and for each $\mathbf{y}^j$ solve the deterministic PDE

$$\nabla \cdot (a(\mathbf{y}^j, x) \cdot \nabla u(\mathbf{y}^j, x)) = f(\mathbf{y}^j, x), \tag{2}$$

using the finite element method to obtain a solution $u_h(\mathbf{y}^j, x)$. Finally, we construct our approximation by Lagrange interpolation:

$$\mathcal{I}_M u_h(\mathbf{y}, x) = \sum_{j=1}^{M} u_h(\mathbf{y}^j, x) \Psi_j(\mathbf{y}) \tag{3}$$

For this scheme, we need to solve M systems of size n_h. For high dimensional spaces Γ, the number of points M needed to obtain a good approximation can be huge!

History of the Multilevel Method

Multilevel methods for SPDEs derive from multigrid methods for the FEM, and have been used most commonly in the context of Monte Carlo methods:

- Multilevel Monte Carlo for numerical integration (S. Heinrich, 2001)
- Multilevel Monte Carlo path simulations for computational finance (M. Giles, 2008)
- Since applied to a variety of SPDEs

History of the Multilevel Method

Multilevel methods for SPDEs derive from multigrid methods for the FEM, and have been used most commonly in the context of Monte Carlo methods:

- Multilevel Monte Carlo for numerical integration (S. Heinrich, 2001)
- Multilevel Monte Carlo path simulations for computational finance (M. Giles, 2008)
- Since applied to a variety of SPDEs

Main Idea: Suppose we have a sequence of finite element solutions $u_{h_k}(\mathbf{y}) \in V_{h_k}$, (with $u_{-1} = 0$). Multilevel methods are based on the following simple identity:

$$u_{h_K}(\mathbf{y}) = \sum_{k=0}^{K} u_{h_k}(\mathbf{y}) - u_{h_{k-1}}(\mathbf{y}).$$

With Monte Carlo methods, we are usually interested in computing some statistics of the approximation $u_{h_K}(\mathbf{y})$. For instance, we can compute expectation using sample averages:

$$\mathbb{E}(u_{h_K}(\mathbf{y})) \approx u_{h_K}^{MLMC} = \sum_{k=0}^{K} \frac{1}{M_{K-k}} \sum_{j=1}^{M_{K-k}} \left(u_{h_k}(\mathbf{y}_j) - u_{h_{k-1}}(\mathbf{y}_j) \right). \quad (4)$$

With Monte Carlo methods, we are usually interested in computing some statistics of the approximation $u_{h_K}(\mathbf{y})$. For instance, we can compute expectation using sample averages:

$$\mathbb{E}(u_{h_K}(\mathbf{y})) \approx u_{h_K}^{MLMC} = \sum_{k=0}^{K} \frac{1}{M_{K-k}} \sum_{j=1}^{M_{K-k}} \left(u_{h_k}(\mathbf{y}_j) - u_{h_{k-1}}(\mathbf{y}_j) \right) . \quad (4)$$

For stochastic collocation, we interpolate the differences at different resolutions. Suppose we have a sequence of interpolation operators $\{\mathcal{I}_{l_k}\}$ with increasing approximation properties. Now the (fully discrete) multilevel approximation is given by:

$$u_{h_K}^{ML}(\mathbf{y}) = \sum_{k=0}^{K} \mathcal{I}_{l_{K-k}} \left(u_{h_k}(\mathbf{y}) - u_{h_{k-1}}(\mathbf{y}) \right) . \quad (5)$$

Error Splitting

We examine the method by considering the discretization errors independently:

$$\|u - u_{h_K}^{ML}\| \leq \|u - u_{h_K}\| + \|u_{h_K} - \mathcal{I}^{\mathrm{ML}} u_{h_K}\| =: I + II.$$

Error Splitting

We examine the method by considering the discretization errors independently:

$$\begin{aligned} \|u - u^{ML}_{h_K}\| &\leq \|u - u_{h_K}\| + \|u_{h_K} - \mathcal{I}^{\mathrm{ML}} u_{h_K}\| \\ &=: I + II. \end{aligned}$$

The term II can be further split apart using the triangle inequality:

$$\begin{aligned} II &= \| \sum_{k=0}^{K} (u_{h_k} - u_{h_{k-1}}) - \mathcal{I}_{l_{K-k}} (u_{h_k} - u_{h_{k-1}}) \| \\ &\leq \sum_{k=0}^{K} \| (\mathrm{I} - \mathcal{I}_{l_{K-k}})(u_{h_k} - u_{h_{k-1}}) \|. \end{aligned}$$

Now to compute the computational cost, we assume that the spatial discretization converges in h as

$$I \leq C_s h_K^{\alpha},$$

and that the stochastic interpolation operators converge according to:

$$\|(\mathrm{I} - \mathcal{I}_{l_{K-k}})(u_{h_k} - u_{h_{k-1}})\| \leq C_I \, M_{K-k}^{-\mu} \, h_k^{\beta},$$

$$\implies II \leq \sum_{k=0}^{K} C_I \, M_{K-k}^{-\mu} \, h_k^{\beta}.$$

Now to compute the computational cost, we assume that the spatial discretization converges in h as

$$I \leq C_s h_K^{\alpha},$$

and that the stochastic interpolation operators converge according to:

$$\|(\mathrm{I} - \mathcal{I}_{l_{K-k}})(u_{h_k} - u_{h_{k-1}})\| \leq C_I \, M_{K-k}^{-\mu} \, h_k^{\beta},$$

$$\Longrightarrow \; II \leq \sum_{k=0}^{K} C_I \, M_{K-k}^{-\mu} \, h_k^{\beta}.$$

Finally, we compute the cost of the multilevel method using the metric

$$Cost = \sum_{k=0}^{K} M_{K-k} \, C_k^{FEM} \approx \sum_{k=0}^{K} M_{K-k} \, h_k^{-\gamma}. \tag{6}$$

Theorem: Cost of the MLSC Method

Under our assumptions, for any $\varepsilon < e^{-1}$ there exists an integer K such that

$$\|u - u_{h_K}^{ML}\|_{L^2_\rho(\Gamma; H^1_0(D))} \leq \varepsilon$$

and

$$Cost_\varepsilon \lesssim \begin{cases} \varepsilon^{-\frac{1}{\mu}}, & \text{if } \beta > \mu\gamma, \\ \varepsilon^{-\frac{1}{\mu}} |\log \varepsilon|^{1+\frac{1}{\mu}}. & \text{if } \beta = \mu\gamma, \\ \varepsilon^{-\frac{1}{\mu} - \frac{\gamma\mu - \beta}{\alpha\mu}}, & \text{if } \beta < \mu\gamma. \end{cases} \tag{7}$$

Theorem: Cost of the MLSC Method

Under our assumptions, for any $\varepsilon < e^{-1}$ there exists an integer K such that

$$\|u - u_{h_K}^{ML}\|_{L^2_\rho(\Gamma;H^1_0(D))} \leq \varepsilon$$

and

$$Cost_\varepsilon \lesssim \begin{cases} \varepsilon^{-\frac{1}{\mu}}, & \text{if } \beta > \mu\gamma, \\ \varepsilon^{-\frac{1}{\mu}} |\log \varepsilon|^{1+\frac{1}{\mu}}. & \text{if } \beta = \mu\gamma, \\ \varepsilon^{-\frac{1}{\mu} - \frac{\gamma\mu - \beta}{\alpha\mu}}, & \text{if } \beta < \mu\gamma. \end{cases} \tag{7}$$

Compare to standard, single level SC:

$$Cost_\varepsilon(SL) \eqsim h^{-\gamma} M \eqsim \varepsilon^{-\gamma/\alpha - 1/\mu}.$$

For some specific examples, $\beta = \alpha$, and so the last line reduces to:

$$Cost_\varepsilon \lesssim \varepsilon^{-\gamma/\alpha}$$

Example Problem:

As an example, we consider the following boundary value problem on either $D = (0,1)$ or $D = (0,1)^2$:

$$-\nabla \cdot (a(\omega, x) \nabla u(\omega, x)) = 1, \qquad \text{for } x \in D,$$
$$u(\omega, x) = 0, \qquad \text{for } x \in \partial D\,.$$

Example Problem:

As an example, we consider the following boundary value problem on either $D = (0,1)$ or $D = (0,1)^2$:

$$
\begin{aligned}
-\nabla \cdot (a(\omega,x)\nabla u(\omega,x)) &= 1, \qquad \text{for } x \in D, \\
u(\omega,x) &= 0, \qquad \text{for } x \in \partial D \,.
\end{aligned}
$$

We take the coefficient a to be of the form

$$
a(\omega,x) = 0.5 + \exp\left[\sum_{n=1}^{N} \sqrt{\lambda_n} b_n(x) Y_n(\omega)\right],
$$

where $\{Y_n\}_{n\in\mathbb{N}}$ is a sequence of independent, uniformly distributed random variables on [-1,1], and $\{\lambda_n\}_{n\in\mathbb{N}}$ and $\{b_n\}_{n\in\mathbb{N}}$ are the eigenvalues and eigenfunctions, reap., of the covariance operator with kernel function $C(x,y) = \exp[-\|x-y\|_1]$.

Results in 10D

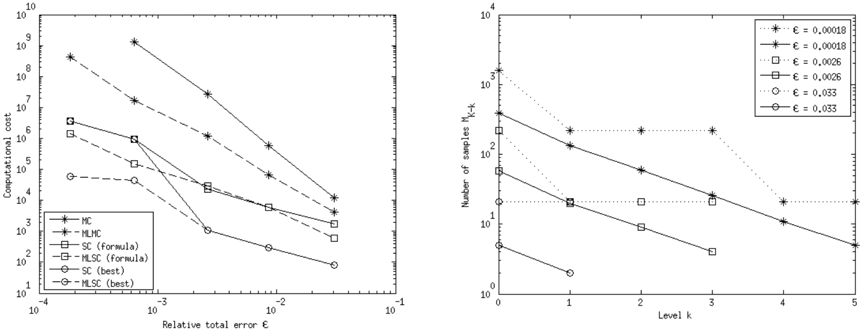

Figure : Left: Cost versus Error for $D = (0,1)^2$, $N = 10$. Right: Number of samples per level (predicted vs actual).

Results in 20D

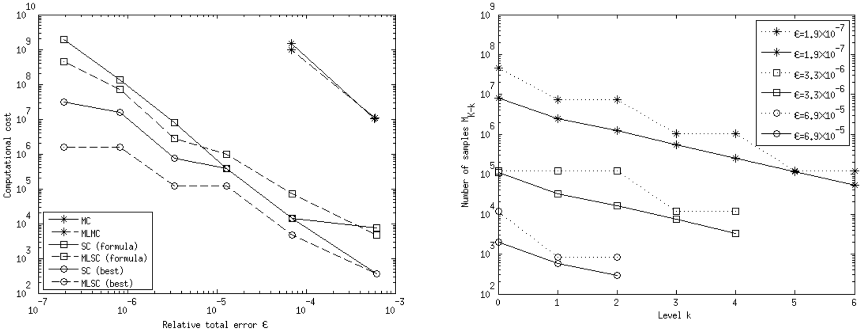

Figure : Left figures: Cost versus Error for $D = (0, 1)$, $N = 20$. Right figures: Number of samples per level (predicted vs actual).

Multilevel methods:

- Can be practically applied to SC methods based on sparse grids
- Reduce computational cost for a variety of stochastic sampling methods for SPDEs.
- Work to counteract the curse of dimensionality.
- Effective when applied to SC schemes even up to 20D.