Mini-course: Uncertainty Quantification and Approximation Theory for Parameterized PDEs

Part VII: multilevel methods for parameterized PDEs

Guannan Zhang[⋆]

Clayton G. Webster[†⋆], Hoang Tran[⋆]

[⋆]Computer Science and Mathematics Division, Oak Ridge National Laboratory
[†]Department of Mathematics, University of Tennessee

Supported by DOE (ASCR), DOD (DARPA), NSF (Comp math)

Vietnam Institute for Advanced Study in Mathematics

November 17, 2016

parameters $\boldsymbol{y} \in \Gamma \subset \mathbb{R}^N$ $\longrightarrow$ **PDE model:** $\mathcal{L}(a(\boldsymbol{y}))[u] = f$ in $D \subset \mathbb{R}^d$, $d = 1, 2, 3$ $\longrightarrow$ quantity of interest $Q[u(\cdot, \boldsymbol{y})]$

- The linear operator $\mathcal{L}$ depends on an q parameters $\boldsymbol{y} = (y_1, y_2, \ldots, y_q) \in \Gamma$, which can be deterministic or stochastic.
- **Deterministic setting**: The parameters $\boldsymbol{y}$ are known or controlled by the user.
  - **Goal**: a query $\boldsymbol{y} \in \Gamma$, quickly approximating the solution map $\boldsymbol{y} \mapsto u(\cdot, \boldsymbol{y})$.
- **Stochastic setting**: The parameters $\boldsymbol{y}$ may be affected by uncertainty (measurement error, incomplete description of parameters), and are modeled as a random vector $\boldsymbol{y} : \Omega \to \Gamma$ with joint PDF $\varrho(\boldsymbol{y}) = \prod_{i=1}^{q} \varrho_i(y_i)$.
  - **Goal**: Uncertainty quantification of u or some statistical QoI, e.g.,

$$\mathbb{E}[u],\ \mathbb{V}\mathrm{ar}[u],\ \mathbb{P}[u > u_0] = \mathbb{E}[\mathbf{1}_{\{u>u_0\}}].$$

- Let $\mathcal{H}(D)$ be a Hilbert space with the norm $\|\cdot\|_{\mathcal{H}}$.
- $\mathcal{B}(\boldsymbol{y})(\cdot,\cdot) : \mathcal{H} \times \mathcal{H} \to \mathbb{R}$ is the bilinear form corresponding to the operator $\mathcal{L}$.
- The weak problem is to find $u : \Gamma \to \mathcal{H}(D)$ s.t.

$$\mathcal{B}(\boldsymbol{y})(u,\phi) = \mathcal{F}(\boldsymbol{y})(\phi), \quad \phi \in \mathcal{H}(D),$$

  where $\mathcal{F}(\boldsymbol{y})(\phi) := (f,\phi) + b(g(\boldsymbol{y}),\phi)$ and $b(\cdot,\cdot)$ is the operator involving the boundary condition.

- The approximation of $\boldsymbol{y} \to u(x,\boldsymbol{y})$ is based on the fact that the manifold $\mathcal{S} := \{u(\cdot,\boldsymbol{y}) : \boldsymbol{y} \in \Gamma\}$ is a compact set in $\mathcal{H}(D)$.
- Galerkin projection to an N-dimensional subspace $\mathcal{H}_N(D) \subset \mathcal{H}(D)$, i.e., find $u_N : \Gamma \to \mathcal{H}_N(D)$ s.t.

$$\mathcal{B}(\boldsymbol{y})(u,\phi) = \mathcal{F}(\boldsymbol{y})(\phi), \quad \phi \in \mathcal{H}_N(D).$$

For any given $\boldsymbol{y} \in \Gamma$, the cost of solving $u_N(x,\boldsymbol{y})$ is dominated by assembling and inverting an $N \times N$ linear system.

Strategies for approximation of the map $\boldsymbol{y} \mapsto u(\cdot, \boldsymbol{y})$

- The first step is to choose a subspace $\mathcal{H}_N(D) \subset \mathcal{H}(D)$, where $\mathcal{H}_N(D)$ is usually an N-dimensional finite element space with N sufficiently large.
- The second step is to approximate the map $\boldsymbol{y} \mapsto u_N(\cdot, \boldsymbol{y})$ by

$$u_N \approx \mathcal{I}_M[u_N](x, \boldsymbol{y}) := \sum_{i=1}^{N} \sum_{j=1}^{M} c_{ij}\, \phi_i(x) \psi_j(\boldsymbol{y}),$$

  - Monte Carlo sampling [1]
  - Stochastic Galerkin
  - Stochastic collocation (interpolation)
  - Discrete least squares
  - Compressed sensing
  - POD, Reduced-basis methods

The main challenge

The computational cost, dominated by the parametric degrees of freedom M, grows very fast with the dimension of the parameter space.

[1] MC method can only approximate statistics, e.g., moments

- Instead of using only one subspace $\mathcal{H}_N$, we use a sequence of nested subspaces

$$\mathcal{H}_{N_0} \subset \mathcal{H}_{N_1} \subset \cdots \subset \mathcal{H}_{N_L} = \mathcal{H}_N \subset \mathcal{H}(D).$$

- $u_N(x, \boldsymbol{y})$ can be represented by as

$$u_N = u_{N_L}(x, \boldsymbol{y}) = \sum_{l=0}^{L} \Delta u_l(x, \boldsymbol{y}),$$

  where $\Delta u_0 := u_{N_0}$ and $\Delta u_l := u_{N_l} - u_{N_{l-1}}$. Then, $u_N(x, \boldsymbol{y})$ is approximated by sampling all $\Delta u_l(x, \boldsymbol{y})$ for $l = 0, \ldots, L$.

- The cost of solving $\Delta u_l(x, \boldsymbol{y})$ is increasing with l, i.e.,

$$\mathcal{C}_0 < \mathcal{C}_1 < \cdots < \mathcal{C}_L.$$

The key idea

The basic idea of multilevel methods is to give a better allocation of computational effort among all the levels of approximations than the single level strategy.

- Multilevel Monte Carlo [Giles, 2015] was originally developed for SDEs, then applied to parameterized PDEs.
- Multilevel quasi-MC [Dick, Kuo, Sloan, 13], Multilevel MCMC [Dodwell, Ketelsen, Scheichl, Teckentrup, 15], etc.
(https://people.maths.ox.ac.uk/gilesm/mlmc_community.html)
- Multilevel stochastic collocation (MLSC) using sparse-grid interpolation [Teckentrup, Jantsch, Webster, Gunzburger, 2015]
- Multi-index methods, including multi-index MC [Haji-Ali, Nobile, Tempone, 2016] and multi-index SC [Haji-Ali, Nobile, Tamellini,Tempone, 2016]
- Multilevel MC and SC methods can be viewed 2-D combination methods, and Multi-index MC and SC can be viewed as multi-D combination methods.

- Monte Carlo method is still a good choice for very high-dimensional UQ problems, when the high-dimensional map $\boldsymbol{y} \to u(\cdot, \boldsymbol{y})$ does NOT have sufficient sparsity to build sparse polynomial approximations.
- Case 1: the random coefficient $a(\boldsymbol{x}, \boldsymbol{y})$ in the parameterized PDE can be approximated by truncated Karhunen-Loève expansion

$$a(\boldsymbol{x}, \boldsymbol{y}) \approx a_{\min} + \exp\left\{\overline{a}(\boldsymbol{x}) + \sum_{k=1}^{d} \sqrt{\lambda_k} b_k(\boldsymbol{x}) y_k(\omega)\right\},$$

  but the random field is so heterogenous that the eigenvalues λ_k decay very slow.

- Case 2: $\boldsymbol{y}$ consists of a large number of independent random parameters that are not from discretization of a random field.
- The PDE model has similar sensitivity with respect to most components of $\boldsymbol{y}$.
- Can we reduce the cost of MC sampling in UQ simulation?

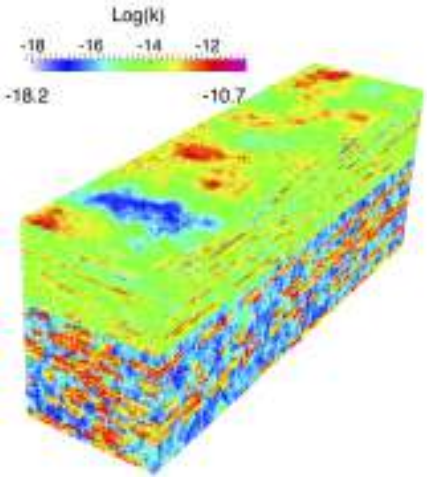
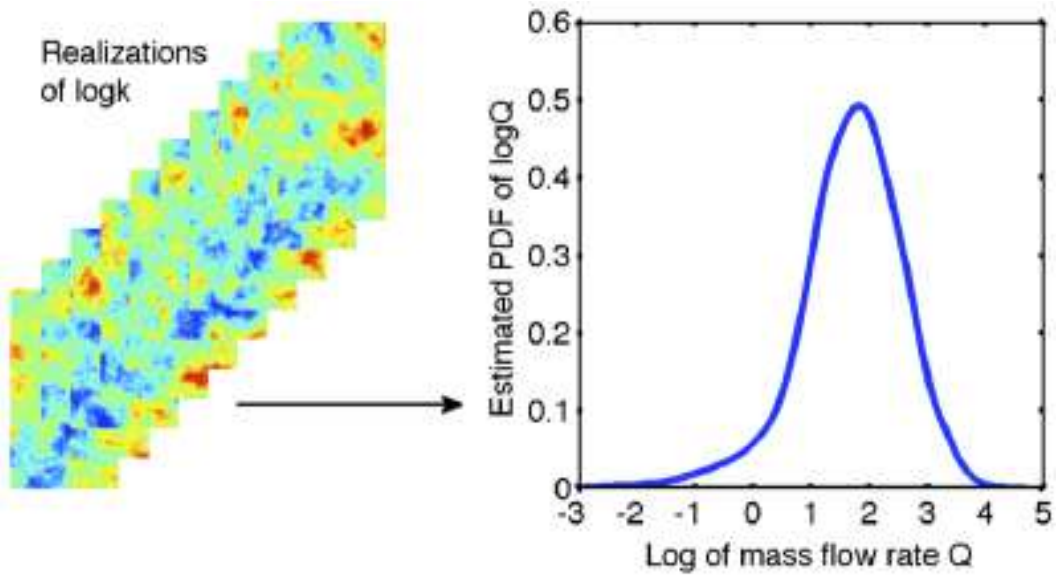


Figure: A highly heterogenous permeability filed (from the SPE 10th model)

- Monte Carlo methods can only be used to do numerical integration, so they cannot be used to approximate the entire solution map $\boldsymbol{y} \to u(\boldsymbol{y})$.
- Quantities of interest: $\mathbb{E}[u]$, $\mathbb{V}\mathrm{ar}[u]$, $\mathbb{P}[u > u_0] = \mathbb{E}[\mathbf{1}_{\{u>u_0\}}]$, CDFs, PDFs.
- Ensemble approximation of the PDF of the quantity of interest $Q(\boldsymbol{y})$

$$\rho(Q) \approx \frac{1}{N} \sum_{n=1}^{N} \left(\sum_{j=1}^{J} \frac{\mathcal{X}_{[e_{j-1},e_j]}(Q(\boldsymbol{y}_n))}{|e_j - e_{j-1}|} \right)$$



- The standard Monte Carlo estimator is

$$\mathbb{E}[Q_M] \approx Q_M^{\mathsf{MC}} = \frac{1}{N_{\mathsf{MC}}} \sum_{i=1}^{N_{\mathsf{MC}}} Q_M^{(i)},$$

and the mean square error (MSE) is

$$e\left(Q_M^{\mathsf{MC}}\right)^2 = \mathbb{E}\left[\left(Q_M^{\mathsf{MC}} - \mathbb{E}[Q]\right)^2\right] = \underbrace{N_{\mathsf{MC}}^{-1}\mathbb{V}[Q_M]}_{\text{sampling error}} + \underbrace{\left(\mathbb{E}[Q_M - Q]\right)^2}_{\text{discretization error}},$$

- The discretization error can be reduced by increasing M, i.e. reducing mesh size.
- The sampling MSE involves the number of samples N_{MC} and the variance of the integrand $\mathbb{V}[Q_M]$.
- The convergence rate $\mathcal{O}(N_{\mathsf{MC}}^{-\frac{1}{2}})$ can be improved to $\mathcal{O}(\log(N_{\mathsf{MC}})^d N_{\mathsf{MC}}^{-1})$ using quasi-MC sampling.
- For a prescribed accuracy $\varepsilon > 0$, reducing the variance $\mathbb{V}[Q_M]$ can also improve the efficiency of achieving $e\left(Q_M^{\mathsf{MC}}\right)^2 \leq \varepsilon^2$.

- $\mathbb{E}[Q_M]$ is estimated using a sequence of models $Q_{M_0}, Q_{M_1}, \ldots, Q_{M_L}$ on a sequence of meshes $\{\mathcal{T}_{M_\ell}\}_{\ell=0}^{L}$, where $M_0 < M_1 < \cdots, M_{L-1} < M_L = M$.
- Denote by $\mathcal{C}_\ell$ the cost of one realization of Q_ℓ for $\ell = 0, 1, \ldots, L$, such that
$$\mathcal{C}_0 > \mathcal{C}_1 > \cdots > \mathcal{C}_L = \mathcal{C}_M$$
- By defining $Y_l := Q_{M_\ell} - Q_{M_{\ell-1}}$, $\mathbb{E}[Q_M]$ can be represented by
$$\mathbb{E}[Q_M] = \mathbb{E}[Q_{M_0}] + \sum_{\ell=1}^{L} \mathbb{E}[Q_{M_\ell} - Q_{M_{\ell-1}}] = \sum_{\ell=0}^{L} \mathbb{E}[Y_\ell],$$
- $\mathbb{E}[Y_0]$ and each incremental item $\mathbb{E}[Y_\ell]$ can be estimated individually by
$$\widehat{Y}_0 := \frac{1}{N_0} \sum_{i=1}^{N_0} Q_0^{(i)} \text{ and } \widehat{Y}_\ell := \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} \left(Q_\ell^{(i)} - Q_{\ell-1}^{(i)} \right),$$
- The quantity of interest $\mathbb{E}[Q_M]$
$$\mathbb{E}[Q_M] \approx Q_M^{\mathsf{ML}} := \sum_{\ell=0}^{L} \widehat{Y}_\ell.$$

- The MSE of the MLMC estimator Q_M^{ML} is

$$e\left(Q_M^{\mathsf{ML}}\right)^2 = \sum_{\ell=0}^{L} N_\ell^{-1}\mathbb{V}[Y_\ell] + \left(\mathbb{E}[Q_M - Q]\right)^2,$$

- The total costs of MC and MLMC are

$$(\mathsf{MC})\ \ \mathcal{C}(Q_M^{\mathsf{MC}}) = N_{\mathsf{MC}}\,\mathcal{C}_L, \quad (\mathsf{MLMC})\ \ \mathcal{C}(Q_M^{\mathsf{ML}}) = \sum_{\ell=0}^{L} N_\ell\,\mathcal{C}_\ell,$$

where the minimum N_{MC} and $\{N_\ell\}_{\ell=0}^{L}$ for achieving a prescribed accuracy $\varepsilon > 0$ is

$$(\mathsf{MC})\ \ N_{\mathsf{MC}} = 2\varepsilon^{-2}\mathbb{V}[Q_M], \quad (\mathsf{MLMC})\ \ N_\ell = 2\varepsilon^{-2}\sqrt{\mathbb{V}[Y_\ell]/\mathcal{C}_\ell}\left(\sum_{\ell=0}^{L}\sqrt{\mathbb{V}[Y_\ell]\mathcal{C}_\ell}\right).$$

- The key idea of MLMC is the fact that $\mathbb{V}[Y_\ell] = \mathbb{V}[Q_{M_\ell} - Q_{M_{\ell-1}}] \to 0$ as $\ell \to \infty$, such that

$$N_0 > N_1 > \cdots > N_L \ll N_{\mathsf{MC}}$$

Under the following three conditions

- Discretization error decay: $|\mathbb{E}[Q_M - Q]| \leq c_1 M^{-\alpha}$
- Variance decay: $\mathbb{V}[Y_\ell] \leq c_2 M_\ell^{-\beta}$,
- Cost increase: $\mathcal{C}_\ell \leq c_3\, M_\ell^{\gamma}$,

the costs of MC and MLMC, for achieving a prescribed accuracy $\varepsilon > 0$, can be represented by

$$\mathcal{C}(Q_M^{\mathsf{MC}}) \leq c_2 N_{\mathsf{MC}} M^{\gamma} \leq 2^{1+\frac{\gamma}{2\alpha}}\, c_1^{\gamma/\alpha}\, c_2 \mathbb{V}[Q_M]\, \varepsilon^{-2-\gamma/\alpha},$$

and

$$\mathcal{C}(Q_M^{\mathsf{ML}}) \leq \begin{cases} c_4 \varepsilon^{-2}, & \text{if } \beta > \gamma, \\ c_4 \varepsilon^{-2} (\log \varepsilon)^2, & \text{if } \beta = \gamma, \\ c_4 \varepsilon^{-2-(\gamma-\beta)/\alpha}, & \text{if } \beta < \gamma, \end{cases}$$

- The variance decay rate β is the most important factor to the computational saving of MLMC method.
- MLMC can be viewed as a variance reduction technique.

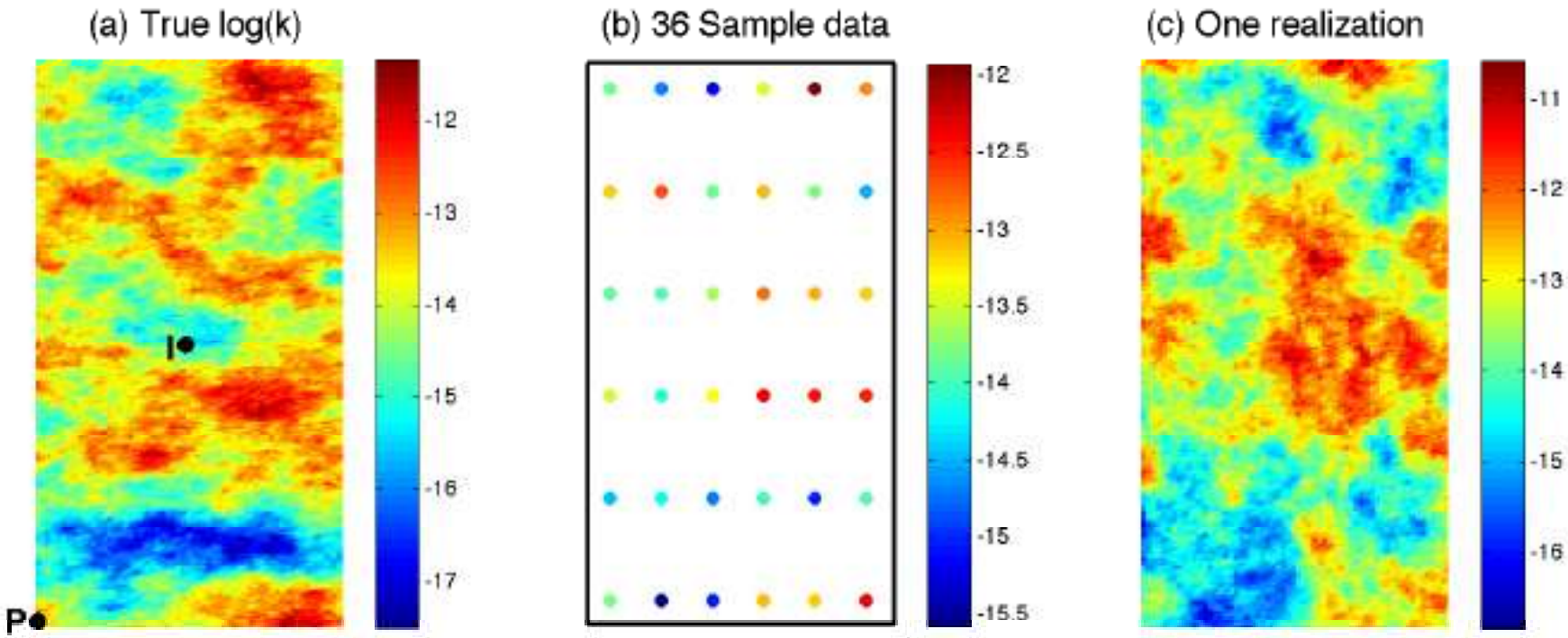


Figure: (a) True log permeability field log(k); I represents the injection well and P is the production well. (b) Locations of 36 sample data drawn from (a); conditioning on these samples, the realizations of random log(k) field are generated. (c) An example of one realization of random log(k) field.

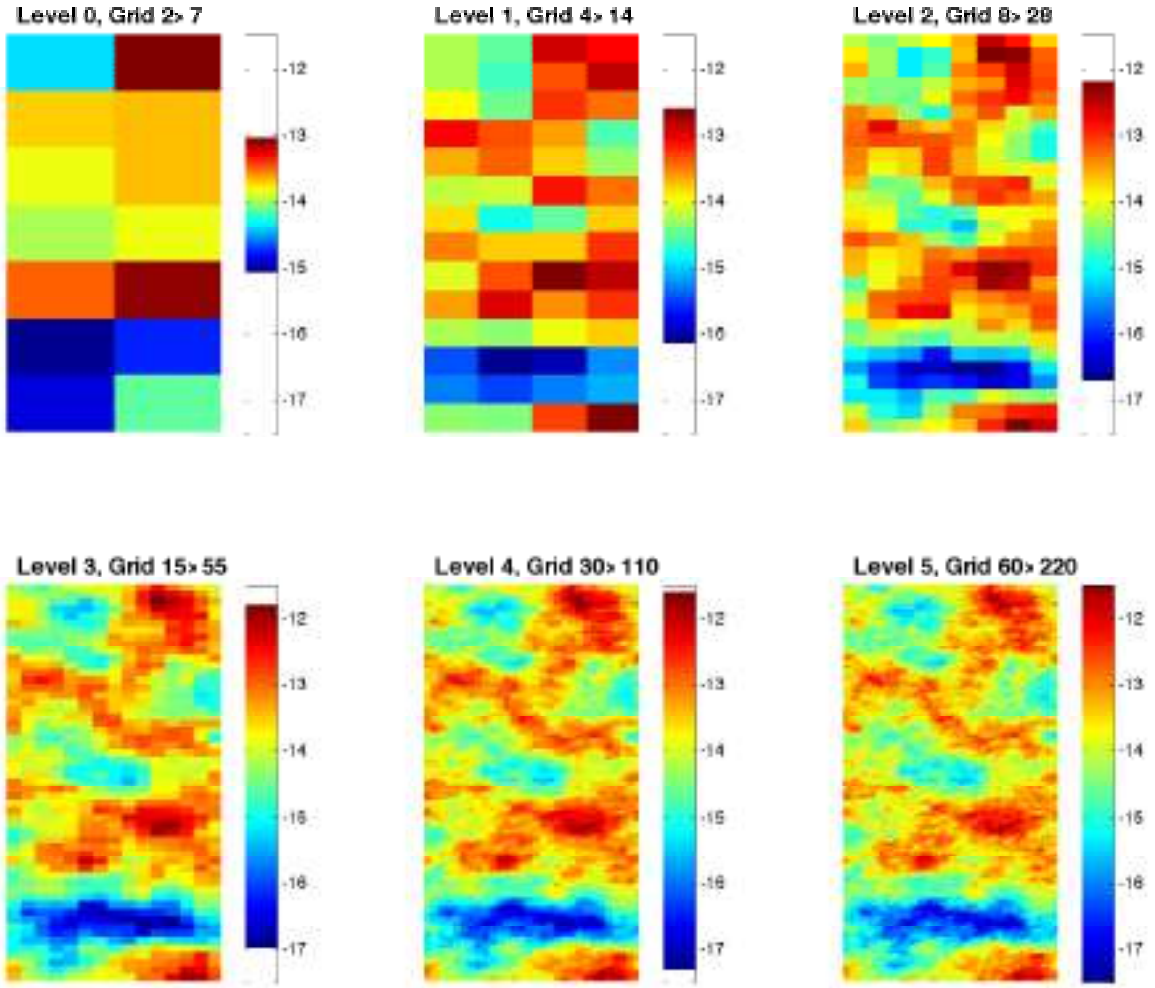


Figure: Grid size of different level grids used in MLMC. The highest level grid (level 5) has the same size at the measurement scale.

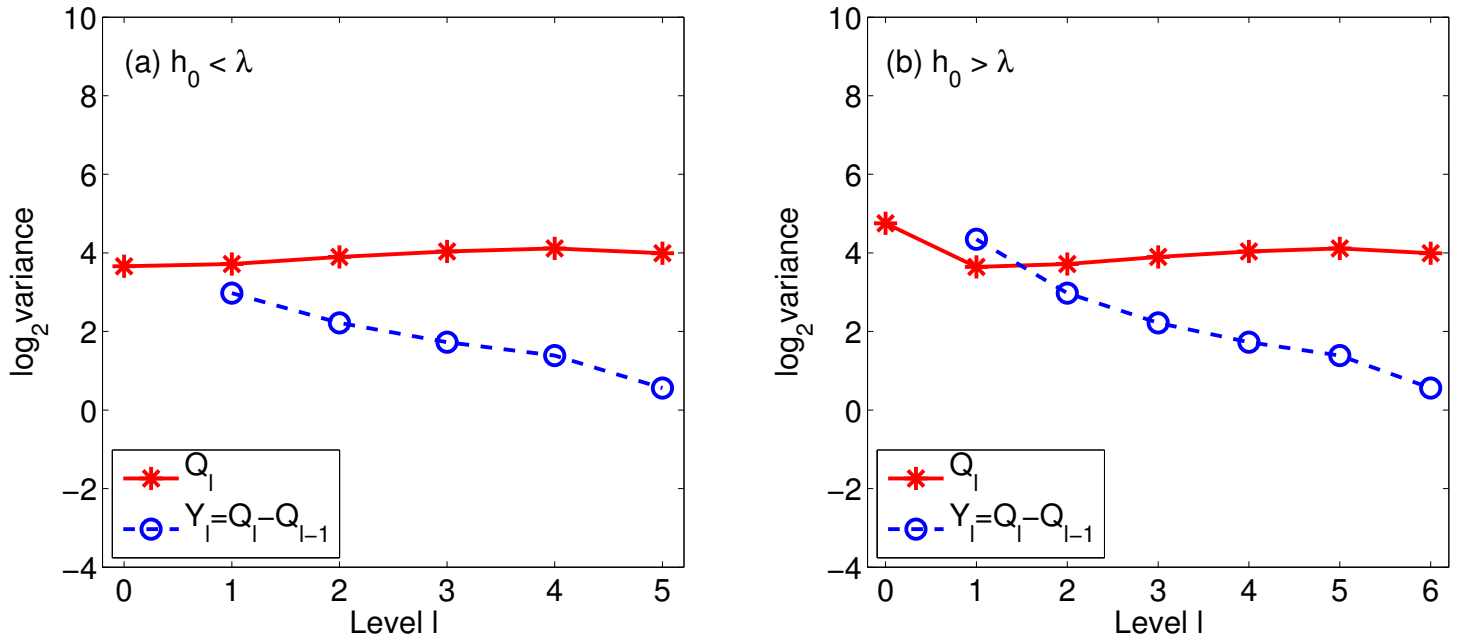
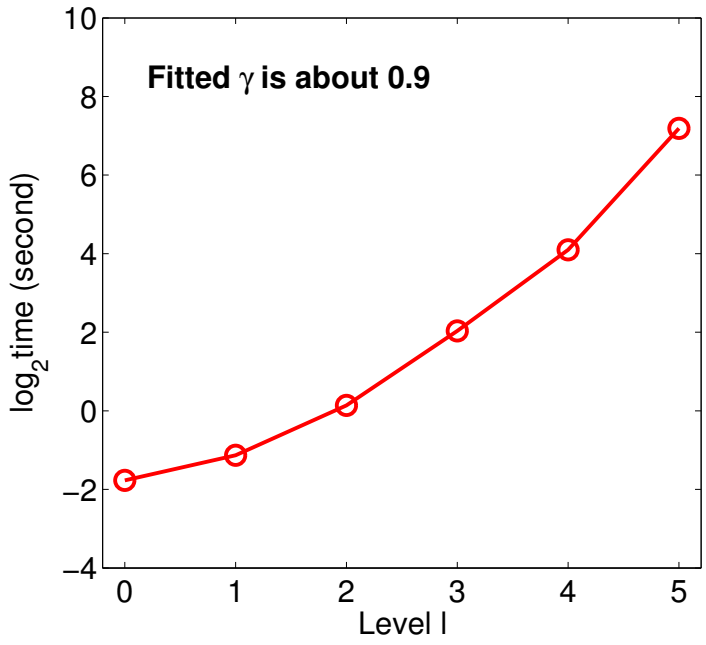
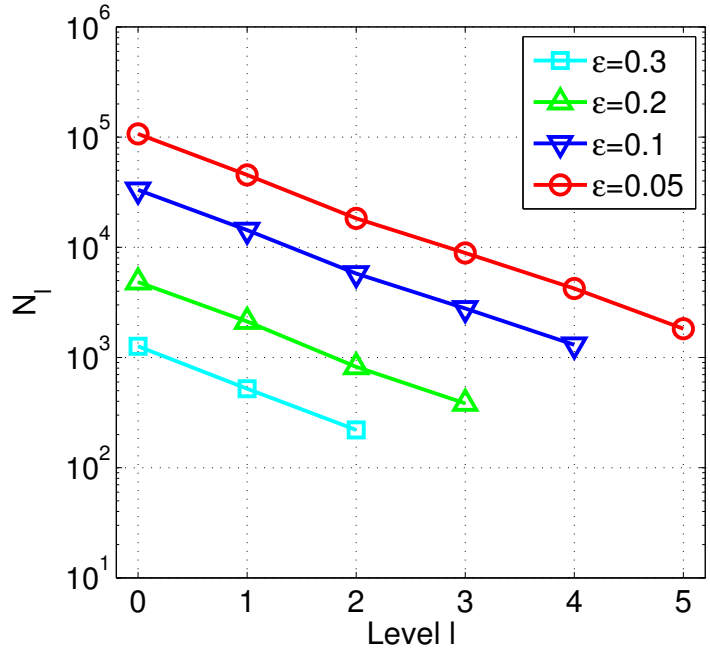
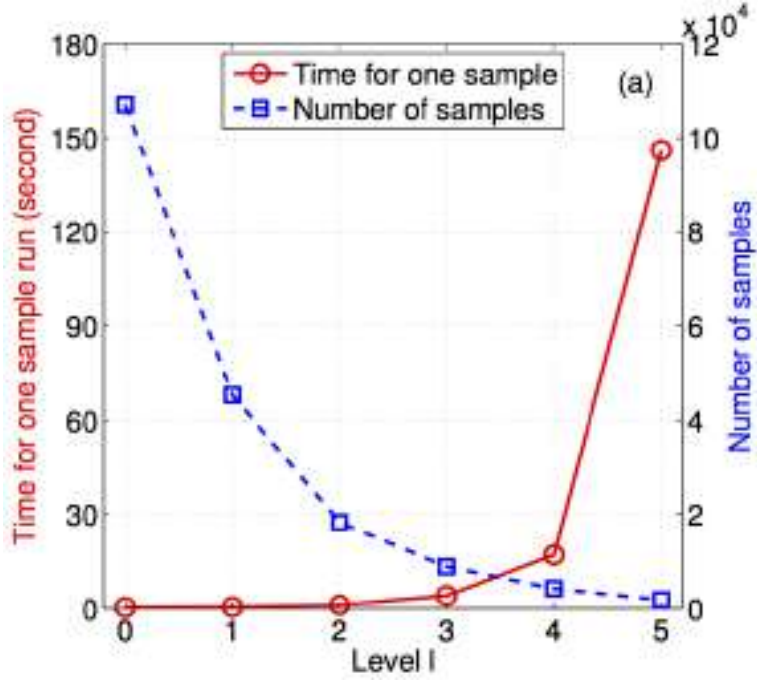
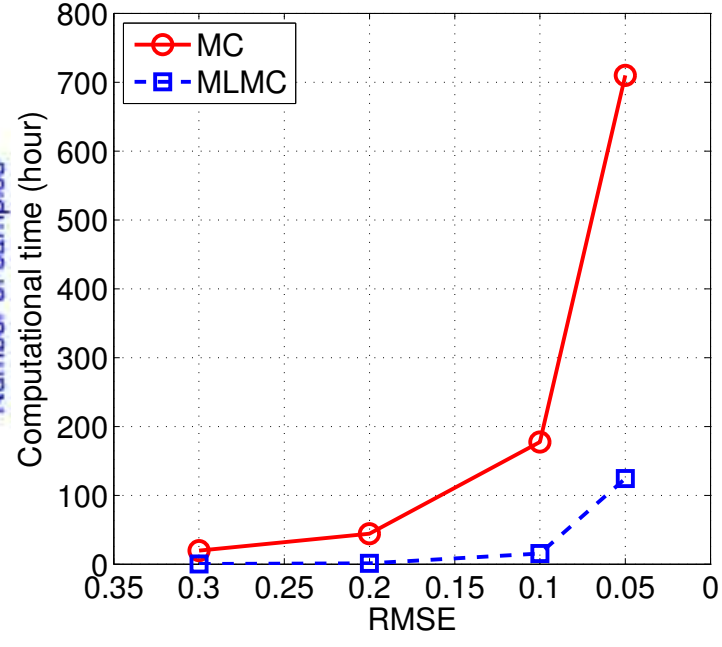


Figure: Plots of $\mathbb{V}[Q_\ell]$ and $\mathbb{V}[Y_\ell] = \mathbb{V}[Q_\ell - Q_{\ell-1}]$ for each level ℓ when (a) $h_0 < \lambda$ and (b) $h_0 > \lambda$ where h_0 is the cell length of the coarsest level grid and λ is the correlation length of the permeability field.


Fitted γ is about 0.9
log2 time (second)
Level l


ε=0.3
ε=0.2
ε=0.1
ε=0.05
N_l
Level l


Time for one sample
Number of samples
(a)
x 10^4
Time for one sample run (second)
Number of samples
Level l

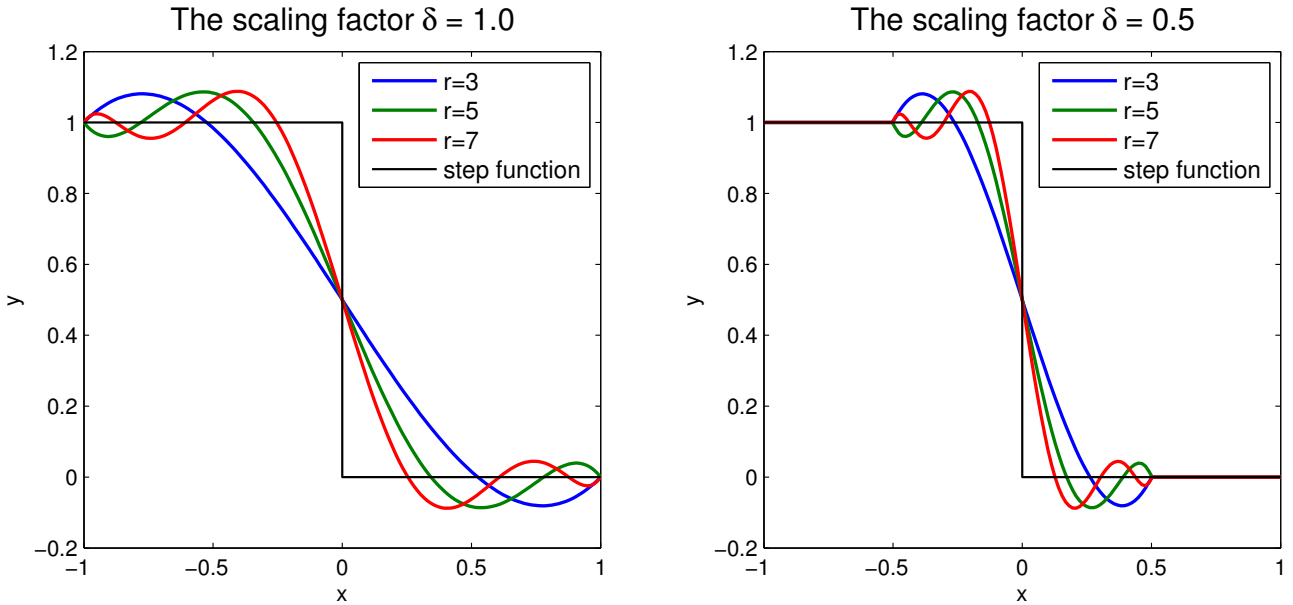

MC
MLMC
Computational time (hour)
RMSE

- The CDF of Q can be represented by

$$F(x) = \mathbb{E}\left[\mathcal{X}_{(-\infty,x]}(Q)\right] = \int_{-\infty}^{+\infty} \mathcal{X}_{(-\infty,x]}(Q)\rho(Q)dQ,$$

  which can be estimated by MC at a set of discrete points $\{x_j\}_{j=1}^{J}$ and Lagrange interpolation.

- The discontinuity of the step function $\mathcal{X}_{(-\infty,x]}(Q)$ makes $\mathbb{V}[Y_\ell]$ NOT decay
- In MLMC setting, smoothing functions are needed to guarantee the required variance decay, i.e. $\mathbb{V}[Y_\ell] \leq c_2 M_\ell^{-\beta}$



- The total errors of MLMC for estimating CDFs can be split as

$$E_{\text{total}} \leq \underbrace{\text{discretization err}}_{E_1} + \underbrace{\text{sampling err}}_{E_2} + \underbrace{\text{interpolation err}}_{E_3} + \underbrace{\text{smoothing err}}_{E_4}$$

- For a prescribed accuracy ε, the goal is to minimize the cost of achieving

$$E_1 \leq \frac{\varepsilon}{4}, \quad E_2 \leq \frac{\varepsilon}{4}, \quad E_3 \leq \frac{\varepsilon}{4}, \quad E_4 \leq \frac{\varepsilon}{4} \quad \Longrightarrow \quad E_{\text{total}} \leq \varepsilon$$

- Observing the fact that

$$E_4 \to 0 \text{ as } \delta \to 0 \text{ but } \mathcal{C}_{\text{total}} \to \infty \text{ as } \delta \to 0.$$

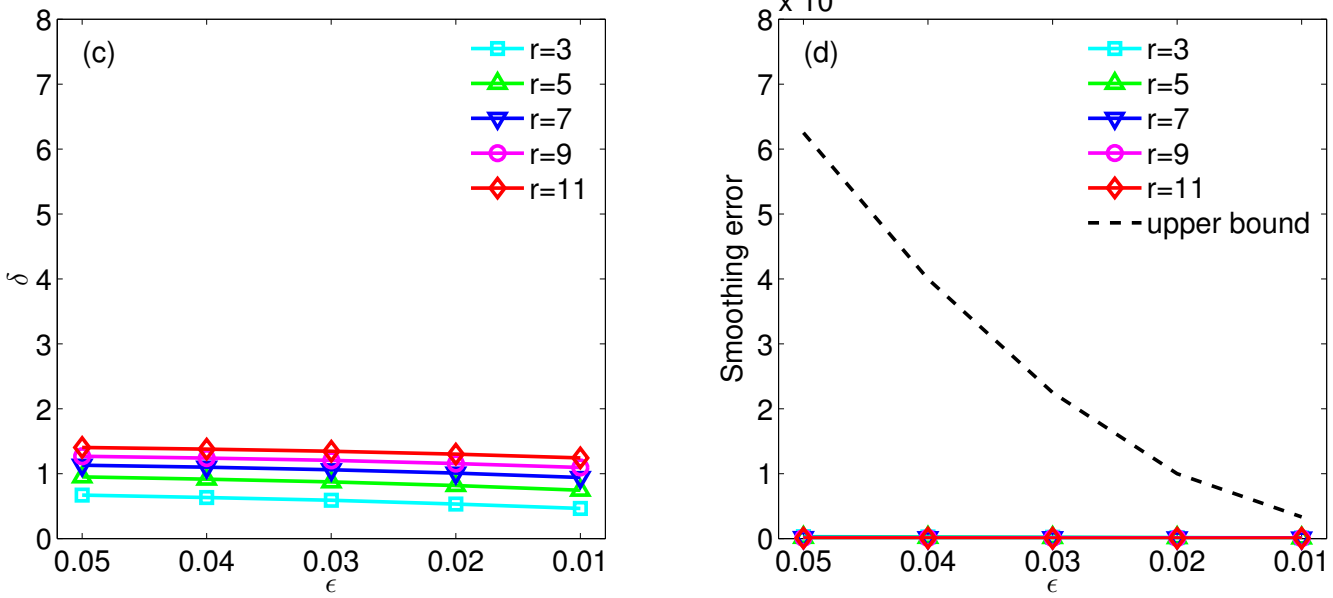
- The challenge is to

Find the maximum value of the scaling factor δ_{opt} to achieve $E_4 \leq \frac{\varepsilon}{4}$ without knowing the representation of the error E_4

- Derive a computable asymptotic estimate of E_4 and find δ using

$$E_4 \leq E_{\text{est}}(\delta) = \frac{\varepsilon}{4} \quad \Longrightarrow \quad \delta = E_{\text{est}}^{-1}\left(\frac{\varepsilon}{4}\right)$$

- However, the estimate $E_{\text{est}}(\delta)$ is usually NOT sharp, such that $E_{\text{est}}^{-1}\left(\frac{\varepsilon}{4}\right)$ may be much smaller than the maximum value δ_{opt} achieving $E_4 \leq \frac{\varepsilon}{4}$.

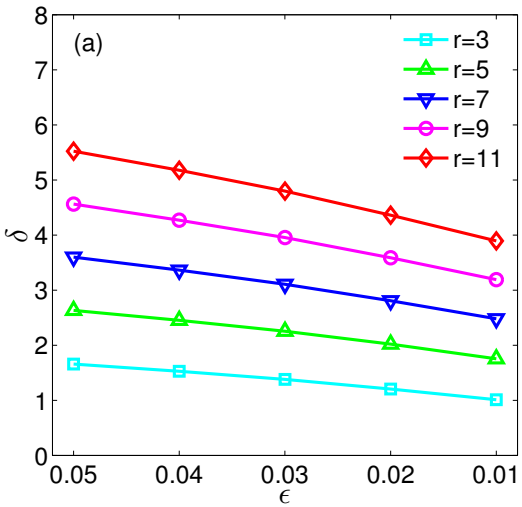


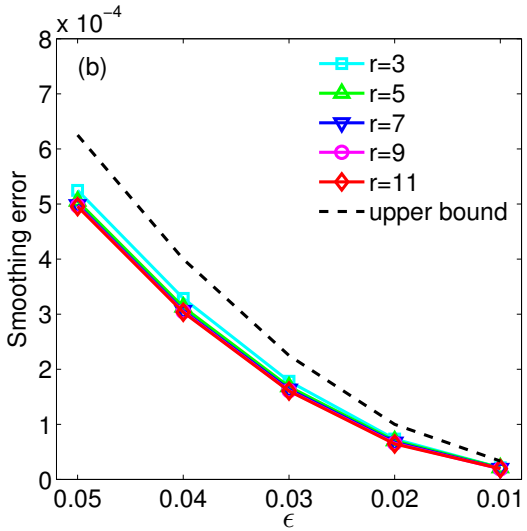
- With too small δ, the cost of MLMC is even higher than MC because the variance decays very slowly.

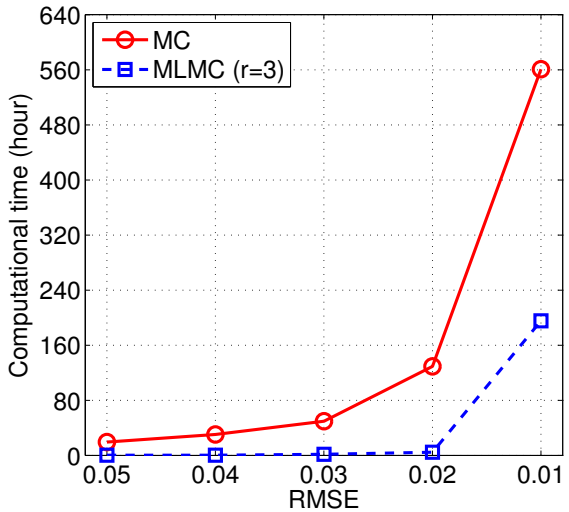
- Draw N_{init} ($N_{\text{init}} \ll N_L$) samples $\{Q^{(i)}\}_{i=1}^{N_{\text{init}}}$, and find the maximum δ, such that

$$\max_{j=1,\ldots,J} \left| \sum_{i=1}^{N_{\text{init}}} \left[\mathcal{X}_{(-\infty,x_j]}(Q^{(i)}) - f_{\text{smooth}}(Q^{(i)}, x_j, \delta) \right] \right|^2 \leq \frac{\varepsilon^2}{16}$$

- The initial samples $\{Q^{(i)}\}_{i=1}^{N_{\text{init}}}$ will be reused in MLMC sampling, so there is no waste of computing effort.







- As a variance reduction technique, the MLMC can significantly reduce the overall computational complexity compared to the standard MC.
- For PDE involved models (e.g. Navier-Stokes equations), the bulk of the computational cost of UQ is associated with linear or nonlinear iterative solvers.
- The convergence of multilevel methods can be further accelerated (and thus, reduce overall complexity) by exploiting the model hierarchy

$$\mathbb{E}[Q_M] = \mathbb{E}[Q_{M_0}] + \sum_{\ell=1}^{L} \mathbb{E}[Q_{M_\ell} - Q_{M_{\ell-1}}] = \sum_{\ell=0}^{L} \mathbb{E}[Y_\ell],$$

- For $\ell = 1, \ldots, L$, the samples of $Q_{M_{\ell-1}}$ can be used to construct good initial guesses for the iterative solvers for solving Q_{M_ℓ}, such that the total number of iterations can be greatly reduced.
  - Multilevel Monte Carlo with acceleration based on mesh-free interpolation [Reshniak-Colgin-Z-Webster,14]
  - Multilevel stochastic collocation with acceleration based on sparse-grid approximation [Galindo-Jantsch-Webster-Z,14]

For stochastic collocation, we simply interpolate the differences $u_{h_k} - u_{h_{k-1}}$ at different resolutions.

Multilevel Stochastic Collocation Method (MLSC)

With general interpolation operators, the multilevel approximation is given by

$$u_K^{ML} = \sum_{k=0}^{K} \mathcal{I}_{M_{K-k}} \left(u_{h_k} - u_{h_{k-1}} \right).$$

Assumption (A1)

Suppose we have a sequence of interpolation operators $\{\mathcal{I}_{M_k}\}$ with algebraic convergence in the number of points M_k:

$$\| v - \mathcal{I}_{M_k}[v] \|_{L^2_\rho(\Gamma; H^1_0(D))} \leq C_I \, M_k^{-\mu} \, \zeta(v),$$

with ζ some function that admits the estimate

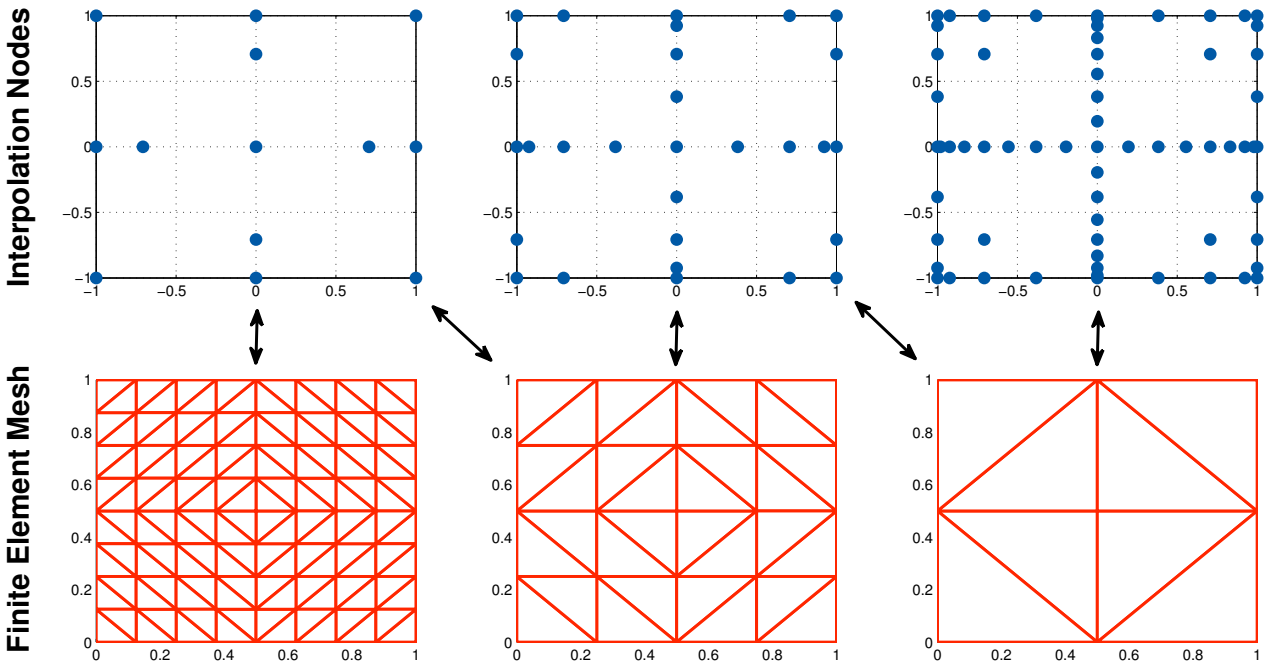
$$\zeta(u_{h_{k+1}} - u_{h_k}) \leq h_{k+1}^{\beta}.$$

$$u_K^{ML} = \sum_{k=0}^{K} \mathcal{I}_{M_{K-k}} \left(u_{h_k} - u_{h_{k-1}} \right), \qquad u_{h_{-1}} = 0.$$

- $\boldsymbol{k = 0}$: $\mathcal{I}_{M_K}[u_{h_0}]$

  $\vdots$

- $\boldsymbol{k = K}$: $\mathcal{I}_{M_0}[u_{h_K} - u_{h_{K-1}}]$



We examine the method by considering the discretization errors independently:

$$\|u - u_K^{ML}\| \leq \underbrace{\|u - u_{h_K}\|}_{I} + \underbrace{\|u_{h_K} - u_K^{\mathrm{ML}}\|}_{II} \leq \varepsilon.$$

Assumption (A2):

The spatial discretization converges in h as

$$I \leq C_s h_K^{\alpha}, \quad \alpha > 0$$

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

$$\begin{aligned} II &= \| \sum_{k=0}^{K} (u_{h_k} - u_{h_{k-1}}) - \mathcal{I}_{M_{K-k}}(u_{h_k} - u_{h_{k-1}}) \| \\ &\leq \sum_{k=0}^{K} \| (\mathsf{I} - \mathcal{I}_{M_{K-k}})(u_{h_k} - u_{h_{k-1}}) \|. \\ &\overset{\text{(A1)}}{\leq} \sum_{k=0}^{K} C_I \, M_{K-k}^{-\mu} \, h_k^{\beta}. \end{aligned}$$

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

$$Cost = \sum_{k=0}^{K} M_{K-k}\, C_k^{FEM} \overset{(A3)}{\simeq} \sum_{k=0}^{K} M_{K-k}\, h_k^{-\gamma}.$$

We can ensure $I \leq \varepsilon/2$ by requiring

$$C_s\, h_K^{\alpha} \leq \frac{\varepsilon}{2}$$

This fixes the mesh maximum mesh size (and the lower level meshes). Then we may choose the interpolation operators $\mathcal{I}_{M_k}$ to satisfy

$$II \leq \sum_{k=0}^{K} C_I\, M_{K-k}^{-\mu}\, h_k^{\alpha} \leq \frac{\varepsilon}{2}.$$

and such that they minimize the computational cost.

Theorem: [Gunzburger, Jantsch, Teckentrup, Webster, 15]

Under assumptions (A1)-(A3), for any $\varepsilon > 0$ there exists an integer K such that

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

and

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

Compare to standard, single level SC:

$$\text{Cost}_\varepsilon^{SL} \eqsim h^{-\gamma} M \eqsim \varepsilon^{-\gamma/\alpha - 1/\mu}.$$

Case	$\beta > \mu\gamma$	$\beta = \mu\gamma$	$\beta = \alpha < \mu\gamma$
Cost Reduction	$\varepsilon^{-\gamma/\alpha}$	$\eqsim \varepsilon^{-\gamma/\alpha}$	$\varepsilon^{-1/\mu}$

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

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

We take the coefficient a to be of the form

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

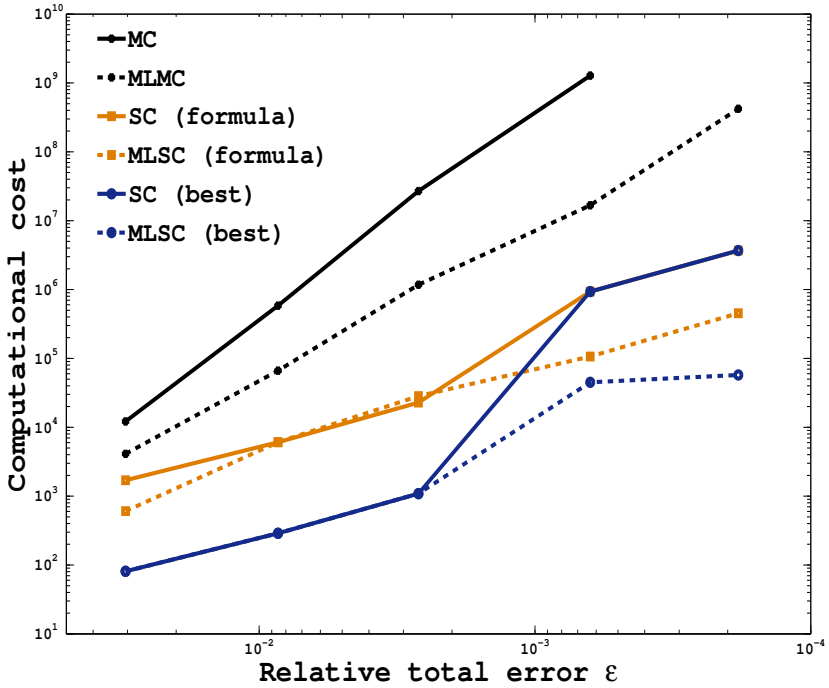
where $\{y_n\}_{n\in\mathbb{N}}$ is a sequence of independent, uniformly distributed random variables on [-1,1], and $\{\lambda_n\}_{n\in\mathbb{N}}$ and $\{b_n\}_{n\in\mathbb{N}}$ are the eigenvalues and eigenfunctions, resp., of the covariance operator with kernel function $C(x_1, x_2) = \exp[-\|x_1 - x_2\|_1]$.

- First, the multilevel method predicts a number of points, but not an appropriate sparse grid. Thus, we may use many more points than necessary.

  We could ameliorate this issue by using granular or adaptive grids, solving a discrete optimization problem for grid levels, or various *ad hoc* rounding strategies. However, we still see gains in efficiency in the "worst" case.

  Leja sequence can be used to achieve linear growth of the sampling in the parameter space.

- Secondly, the method relies on *a priori* knowledge of the convergence rates and parameters $\alpha, \beta, \mu, \gamma$. We need a practical implementation of the method. (See *MLMC path simulation*, Giles, 2008).



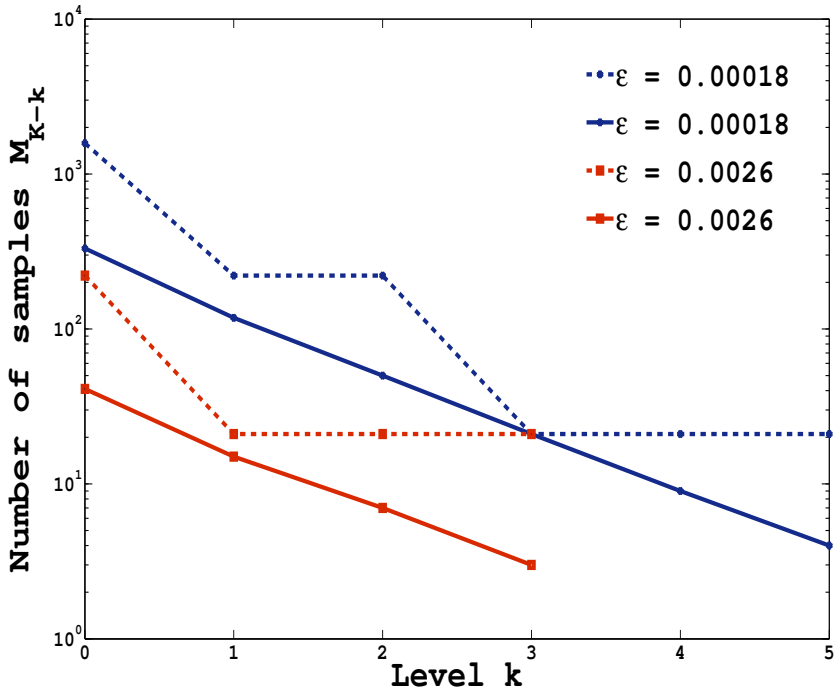
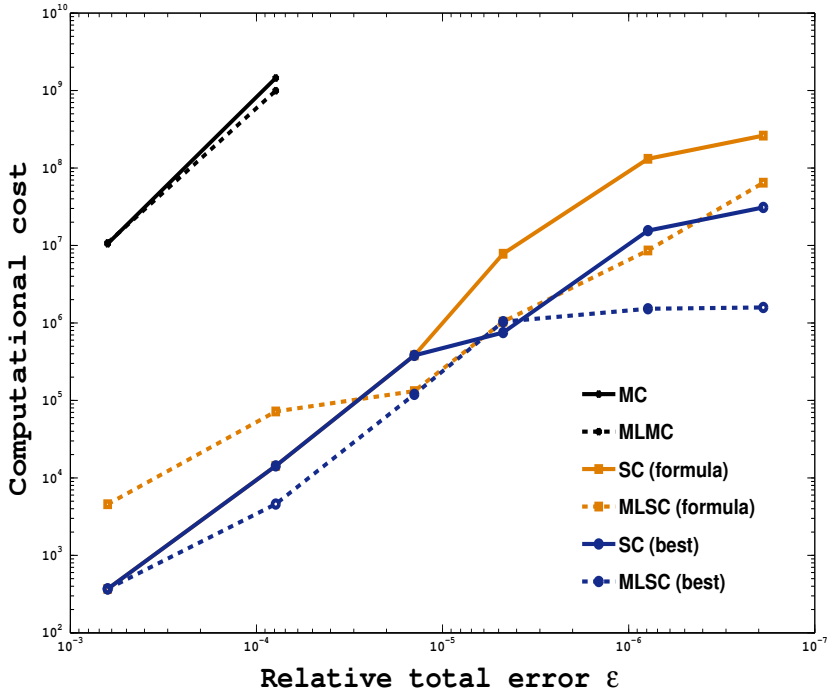


Figure: Left: Cost versus Error for $D = (0, 1)^2$, $N = 10$. Right: Number of samples per level (predicted vs actual).

Results in N=20D



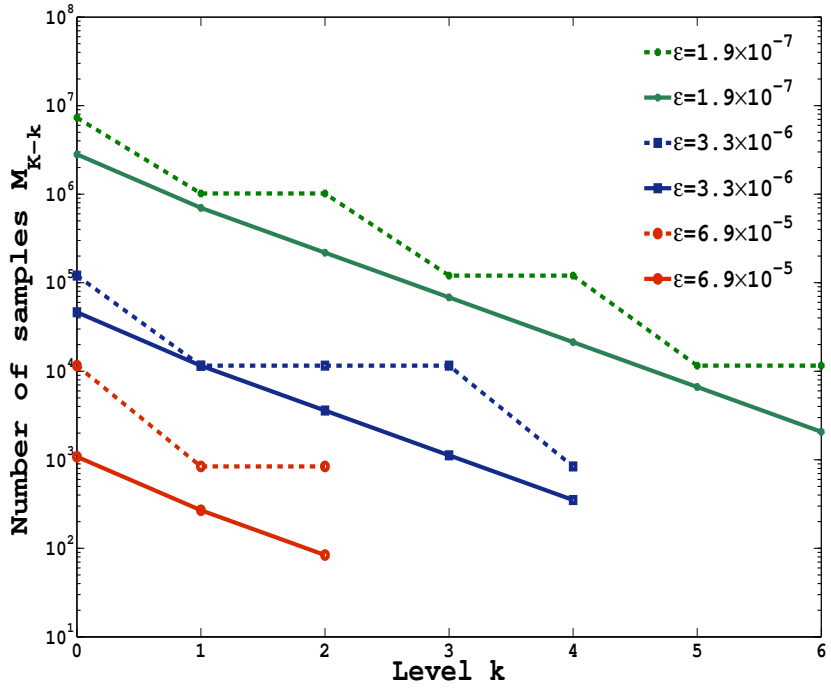


Figure: Left: Cost versus Error for $D = (0, 1)$, $N = 20$. Right: Number of samples per level (predicted vs actual).

The effectiveness of a multilevel strategy relies on the following assumptions:

A_1 FEM error: $\|u_{N_l} - u\|_{\mathcal{H}} \leq c_1 N_l^{-\alpha}$,

A_2 Parametric approximation error: $\|\Delta u_l - \mathcal{I}_{M_l}[\Delta u_l]\| \leq c_2\, \sigma(M_l)\, N_l^{-\beta}$,

A_3 Cost increase: $\mathcal{C}_\ell \leq c_3\, N_l^{\gamma}$,

for some positive constants $\alpha, \beta, \gamma > 0$.

- Multilevel Monte Carlo (MLMC):
  - Pro: linear growth of the number of snapshots (samples).
  - Con: slow convergence; only for approximating statistics, e.g. moments.
- Multilevel stochastic collocation (MLSC):
  - Pro: fast convergence; approximation to the solution map $\boldsymbol{y} \mapsto u(\cdot, \boldsymbol{y})$.
  - Con: fast growth of interpolation points; structured interpolation grid.

Motivation

The reduced-basis method features both fast convergence as MLSC and unstructured sampling as MLMC.

- Choose a training set Ξ_{train} of M_{train} points in the parameter domain Γ
  - e.g., these points could be chosen randomly according to the joint PDF $\rho(\boldsymbol{y})$ associated with the random parameters $\boldsymbol{y} \in \Gamma$
- Choose a point $\boldsymbol{y}_0 \in \Xi_{\text{train}}$
  - the particular point chosen is usually somewhere near the center of Γ
- Then, starting with $n = 1$, we do the following steps
- Solve the spatial finite element system n times to obtain the set of n finite element approximations $\{u_N(x, \boldsymbol{y}_i)\}_{i=0}^{n-1}$
  - let $\mathcal{H}_n^{\text{RB}} = \text{span}\{u_N(x, \boldsymbol{y}_i)\}_{i=0}^{n-1} \subset \mathcal{H}_N \quad \Longleftarrow$ reduced basis space of dimension $n \ll N$
- Using a Galerkin method, determine a reduced basis approximate solution $u_n^{\text{RB}}(x, \boldsymbol{y}) \in \mathcal{H}_n^{\text{RB}}(D)$, i.e., solve the problem by seeking

$$u_n^{\text{RB}}(x, \boldsymbol{y}) = \sum_{i=0}^{n-1} c_i^k(\boldsymbol{y}) u_N(x, \boldsymbol{y}_i) \in \mathcal{H}_n^{\text{RB}}(D)$$

satisfying

$$\mathcal{B}(\boldsymbol{y})(u_n^{\text{RB}}, \phi) = \mathcal{F}(\boldsymbol{y})(\phi), \quad \phi \in \mathcal{H}_n^{\text{RB}}(D)$$

- The next step is to "improve" the initial point set $\Xi_n = \{\boldsymbol{y}_i\}_{i=0}^{n-1}$

  $\Longrightarrow$ add a point $\boldsymbol{y}_n$ to the set Ξ_n
  to produce
  a "better" point set $\Xi_{n+1} = \{\boldsymbol{y}_i\}_{i=0}^{n}$ of $n+1$ points

- Ideally, the new point $\boldsymbol{y}_n$ is chosen by the greedy algorithm

$$\boldsymbol{y}_n = \arg\sup_{\boldsymbol{y}\in\Gamma} |||u_n^{\mathrm{RB}}(\cdot,\boldsymbol{y}) - u_N(\cdot,\boldsymbol{y})|||$$

  $\Longrightarrow$ the point in Γ at which the error in the current reduced basis approximation relative to the finite element solution is largest.

- Determine the finite element solution $u_N(x,\boldsymbol{y}_n)$ corresponding to the new point $\boldsymbol{y}_n$

- set $\Xi_{n+1} = \Xi_n \cup \{\boldsymbol{y}_n\}$ and define

$$\mathcal{H}_{n+1}^{\mathrm{RB}} = \mathrm{span}\{u_N(x,\boldsymbol{y}_i)\}_{i=0}^{n} \subset \mathcal{H}_N \quad \Longleftarrow$$

  new reduced basis space of dimension $n+1 \ll N$

- Obvious flaw: Determining the point $\boldsymbol{y}_n$ by solving the maximization problem

$$\boldsymbol{y}_n = \arg\sup_{\boldsymbol{y}\in\Gamma} |||u_n^{\mathrm{RB}}(\cdot,\boldsymbol{y}) - u_N(\cdot,\boldsymbol{y})|||$$

  is impossible.

- Instead, we could solve the discretized problem

$$\boldsymbol{y}_n = \arg\max_{\boldsymbol{y}\in\Xi_{\mathrm{train}}} |||u_n^{\mathrm{RB}}(\cdot,\boldsymbol{y}) - u_N(\cdot,\boldsymbol{y})|||$$

  Since the initial point set Ξ_n is chosen from the training set, $\Xi_{n+1} \subset \Xi_{\mathrm{train}}$ holds as well.

- Unfortunately, even this is not a practical procedure because it requires M_{train} expensive finite element solves, something we want to avoid.

- We use the Galerkin residual as the error indicator and exploit the affine property of the coefficient $a(x,\boldsymbol{y})$ to achieve the online-offline decomposition.

- The multilevel RB approximation is represented by

$$\mathcal{I}_L^{\mathrm{RB}}[u](x,\boldsymbol{y}) := \sum_{l=0}^{L} \mathcal{I}_{M_l}^{\mathrm{RB}}[\Delta u_l](x,\boldsymbol{y}),$$

  where $\mathcal{I}_{M_l}^{\mathrm{RB}}[\cdot]$ is the classic RB operator acting on Δu_l.

- For any $\boldsymbol{y} \in \Gamma$, $\Delta u_l(x,\boldsymbol{y})$ is obtained by solving

$$\begin{aligned} \mathcal{B}(\boldsymbol{y})(u_{l-1},\phi) &= \mathcal{F}(\boldsymbol{y})(\phi), && \text{for all } \phi \in \mathcal{H}_{N_{l-1}}(D), \\ \mathcal{B}(\boldsymbol{y})(\Delta u_l,\phi) &= \mathcal{F}(\boldsymbol{y})(\phi) - \mathcal{B}(\boldsymbol{y})(u_{l-1},\phi), && \text{for all } \phi \in \mathcal{H}_{N_l}(D), \end{aligned}$$

  where $\|\Delta u_l\|_{\mathcal{H}} \propto N_l^{-\alpha}$, e.g., in $\mathbb{R}^2$ using quadratic FEM yields $\alpha = \frac{3}{2}$.

- Given $\mathcal{H}_n = \mathrm{span}\{\Delta u_l(x,\boldsymbol{y}_1), \Delta u_l(x,\boldsymbol{y}_2), \cdots, \Delta u_l(x,\boldsymbol{y}_n)\}$, the next sample $\boldsymbol{y}_{n+1}$ can be determined by

$$\boldsymbol{y}_{n+1} = \operatorname*{argmax}_{\boldsymbol{y}\in\Gamma} \inf_{\phi\in\mathcal{V}_n} \|\Delta u_l(x,\boldsymbol{y}) - \phi\|_{\mathcal{H}}.$$

- The Kolmogorov n-width for u

$$d_n := \inf_{\dim(\mathcal{H})=\mathrm{n}} \max_{\boldsymbol{y}\in\Gamma} \inf_{\phi\in\mathcal{H}} \|u-\phi\|_{\mathcal{H}}.$$

- The convergence rate of the greedy algorithm is determined by the decay of d_n [Binev, Cohen, Dahmen, DeVore, Petrova, Wojtaszczyk, 2011]

- If d_n exhibits algebraic decay, then

$$\max_{\boldsymbol{y}\in\Gamma} \|u-u_n\|_{\mathcal{H}} \leq C\, d_n \leq C(r) n^{-r}.$$

- The MLRB method requires that the n-width of Δu_l

$$d_{l,n} := \inf_{\dim(\mathcal{H})=\mathrm{n}} \max_{\boldsymbol{y}\in\Gamma} \inf_{\phi\in\mathcal{H}} \|\Delta u_l-\phi\|_{\mathcal{H}},$$

  also decays with N_l, i.e.,

$$d_{l,n} \leq C(l,r) N_l^{-\beta} n^{-r_l},\ l=0,\ldots,L.$$

- Instead of sampling from the entire parameter domain Γ, we sample in a training set

$$\Xi_{\text{train}} := \{\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_s\},$$

  which is uniformly sampled from Γ.

- The Galerkin residual is used as the error indicator to generate the next snapshot

$$\boldsymbol{y}_{n+1} = \underset{\boldsymbol{y}\in\Xi_{\text{train}}}{\text{argmax}} \sup_{\phi\in\mathcal{V}_n} \frac{|\mathcal{F}_l(\boldsymbol{y})(\phi) - \mathcal{B}(\boldsymbol{y})(\Delta u_{l,n}, \phi)|}{\|\phi\|_{\mathcal{H}}}$$

- If the bilinear form $\mathcal{B}(\boldsymbol{y})(\cdot,\cdot)$ is separable, i.e.,

$$\mathcal{B}(\boldsymbol{y})(u,v) = \sum_{k=1}^{m} p_k(\boldsymbol{y})\mathcal{B}_k(u,v),$$

  the cost of evaluating the Galerkin residual only depends on n.

- If $\mathcal{B}(\boldsymbol{y})$ is NOT separable, empirical interpolation [Barrault, Maday, Nguyen, Patera, 04] is needed.

The level l for each snapshot is chosen by the weighted Galerkin residuals.

Define $\mathcal{H}_{N_1}, \mathcal{H}_{N_2}, \cdots$, where $N_l = 4^{l-1} N_1$ by assuming $D \subset \mathbb{R}^2$

- Let $\boldsymbol{y}_1 = \frac{1}{|\Gamma|} \int_\Gamma \boldsymbol{y} d\boldsymbol{y}$
- Let $\mathcal{H}_{n_1} = \text{span}\{u_{N_1}(x, \boldsymbol{y}_1)\}$ and $n_1 = 1$;
- Given $\mathcal{H}_{n_1}, \cdots, \mathcal{H}_{n_L}$, compute the maximum Galerkin residuals $e_1, \cdots, e_L$;
- Next sample is collected at level l^* where

$$l^* = \underset{l \in [1,L]}{\text{argmax}} \ \frac{e_l}{N_l}$$

- Update the RB approximation at level l^*.

Note: Our strategy is closely related to the profit-index approach [Bungartz, Griebel, 04], [Nobile, Tamellini, Tempone, 14] in the context of sparse grid interpolation and integration.

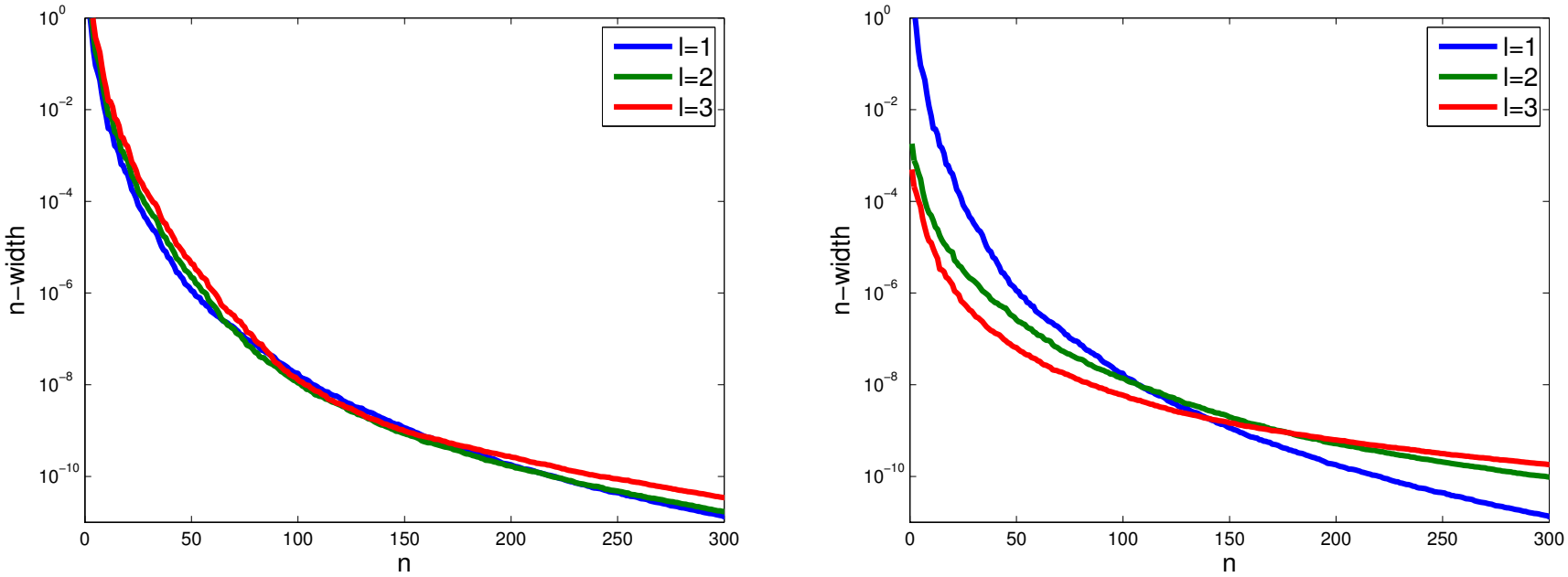
We consider the classical parameterized linear elliptic equation

$$\begin{cases} -\nabla \cdot (a(x, \boldsymbol{y}) \nabla u(x, \boldsymbol{y})) = f(x), & x \in D, \\ u(x, \boldsymbol{y}) = 0, & x \in \partial D, \end{cases}$$

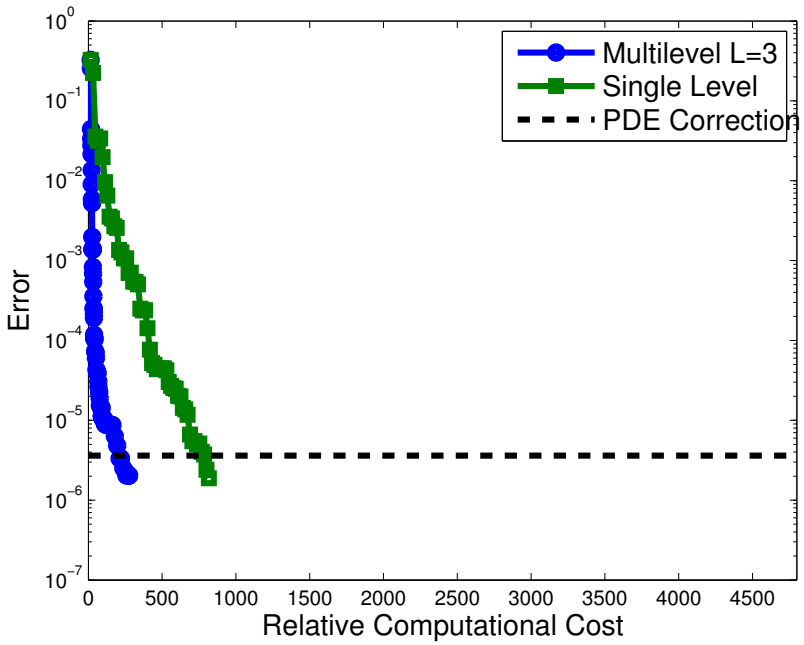
where

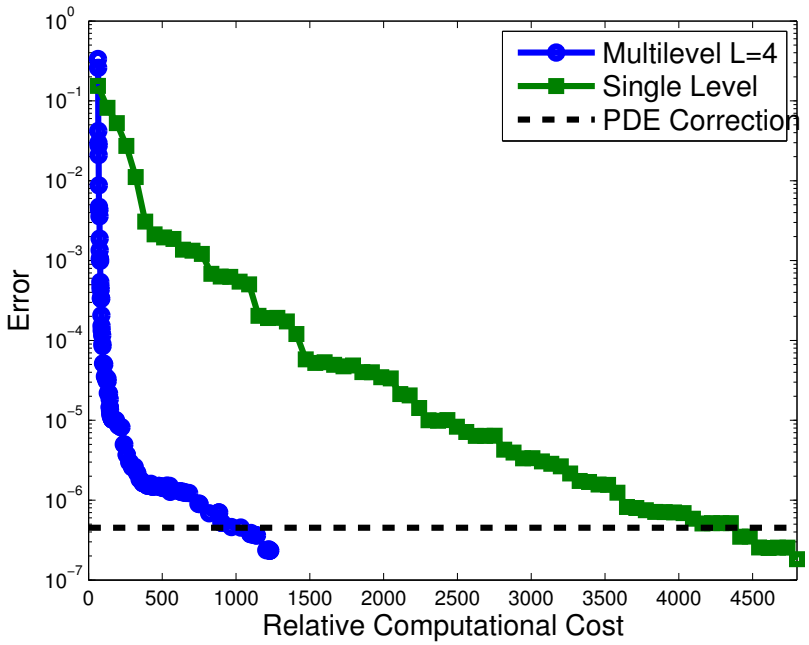
- $D = [0, 1] \otimes [0, 1]$
- $f(x) = \cos(x_1) \sin(x_1)$
- $\boldsymbol{y} \in \Gamma = \bigotimes_{k=1}^{11} [-1, 1]$
- $a(x, \boldsymbol{y})$ is defined by

$$a(x, \boldsymbol{y}) = 0.5 + \exp\left(1 + \frac{\sqrt[4]{9\pi}}{2} y_1 + \frac{\sqrt[4]{9\pi}}{\sqrt{2}} \sum_{k=1}^{5} e^{-\frac{(k\pi)^2}{32}} \left(y_{2k} \sin(k\pi x_1) + y_{2k+1} \cos(k\pi x_1)\right)\right).$$



- The decay of the Kolmogorov n-width for u_l (left) and Δu_l (right)





	n_1	n_2	n_3	n_4	Single-level
$L = 3$	65	27	5	–	49
$L = 4$	138	74	28	8	69

We consider the linear convection diffusion equations

$$\begin{cases} -\nabla \cdot (a(x,\boldsymbol{y})\nabla u(x,\boldsymbol{y})) + b(\boldsymbol{y}) \cdot \nabla u(x,\boldsymbol{y}) &= 0, & x \in D, \\ u(x,\boldsymbol{y}) &= g(x,\boldsymbol{y}), & x \in \partial D, \end{cases}$$

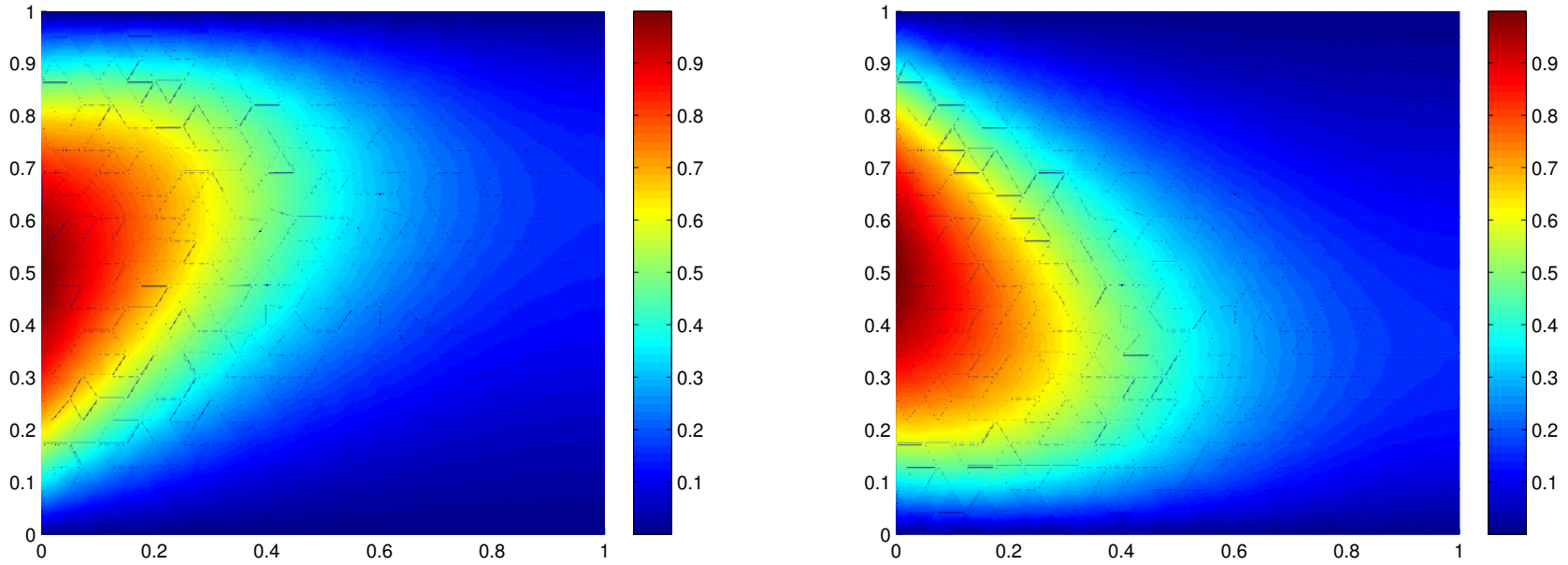
where

- $D = [0,1] \otimes [0,1]$
- $\boldsymbol{y} \in \Gamma = \bigotimes_{k=1}^{6} [-1,1]$
- the parameterized inputs $a(x,\boldsymbol{y})$, $b(\boldsymbol{y})$ and $g(x,\boldsymbol{y})$ are given by

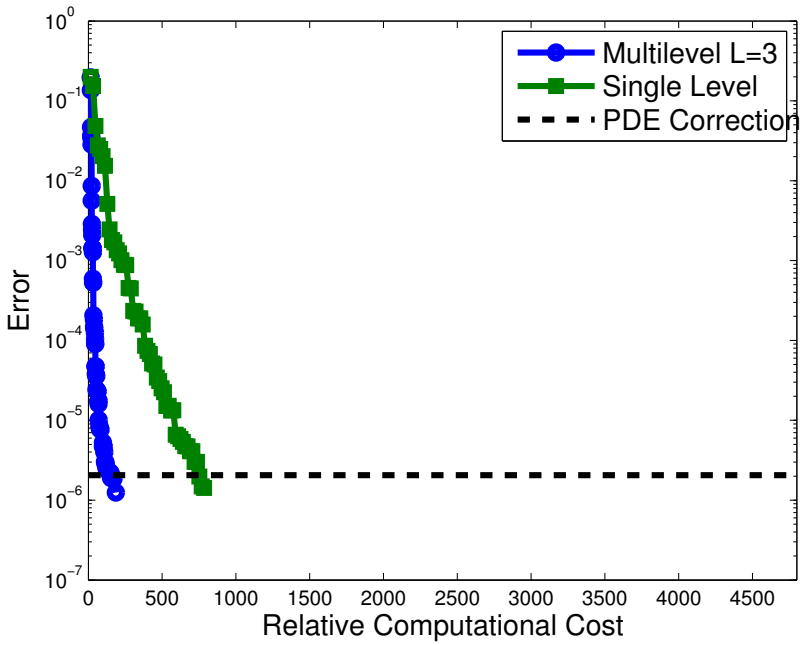
$$a(x,\boldsymbol{y}) = 0.4 + 0.1y_1 + 0.1y_2\cos(\pi x_1) + 0.1y_3\sin(\pi x_1)$$

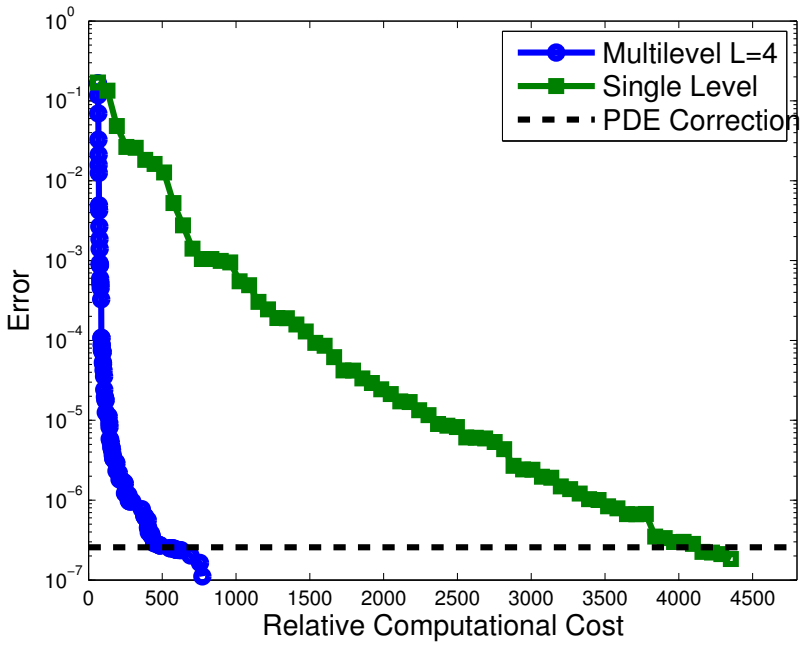
$$b(\boldsymbol{y}) = (1 + 0.25y_4)\begin{pmatrix} \cos(\frac{\pi}{4}y_5) \\ \sin(\frac{\pi}{4}y_5) \end{pmatrix}$$

$$g(x,\boldsymbol{y}) = \begin{cases} 4(0.75 + 0.25y_6)x_2(1-x_2), & x_1 = 0 \\ 0, & x_2 = 0 \text{ or } x_2 = 1 \end{cases}$$



Two realization of the PDE.





	n_1	n_2	n_3	n_3	Single-level
$L=3$	55	21	3	–	48
$L=4$	95	52	22	5	68

- Multilevel methods are very easy to implement, especially the MLMC.
- The key point to guarantee the complexity reduction is to have as much variance reduction as possible;
- However, it is usually not easy to have when the quantity of interest has some irregular dependence on the parameter, e.g., in CDF approximation.
- Multilevel methods can be extended by using a sequence of multi-fidelity models, obtained by upscaling or homogenization.

- M. Barrault, Y. Maday, N. Nguyen, and A. Patera, *An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations*, C. R. Math. Acad. Sci. Paris Series I, 339 (2004), pp. 667-672.
- A. Cohen and R. DeVore, *Approximation of high-dimensional parametric PDEs*, Acta Numerica, (24) 2015, pp. 1–159.
- P. Binev, A. Cohen, W. Dahmen, R. DeVore, G. Petrova, P. Wojtaszczyk, *Convergence Rates for Greedy Algorithms in Reduced Basis Methods*, SIAM Journal on Mathematical Analysis 43 (2011), 1457-1472.
- M. Giles. *Multilevel Monte Carlo methods*. Acta Numerica, 24:259-328, Cambridge University Press, 2015
- A.L. Teckentrup, P. Jantsch, C.G. Webster, M. Gunzburger. *A multilevel stochastic collocation method for PDEs with random input data*, SIAM/ASA Journal on Uncertainty Quantification, 3(1):1046-1074, 2015.
- Dick, J., Kuo, F.Y. and Sloan, I.H. *High-dimensional integration: The quasi-Monte Carlo way*, Acta Numerica, Vol. 22, pp. 133–288.
- M.B. Giles. *Multilevel Monte Carlo path simulation*, Operations Research, 56(3):607-617, 2008.
- M. Giles, T. Nagapetyan, K. Ritter. *Multi-Level Monte Carlo approximation of distribution functions and densities*. SIAM/ASA Journal on Uncertainty Quantification, 3:267-295, 2015.
- D. Lu. G. Zhang, C. Webster, C. Barbier, *An improved multilevel Monte Carlo method for estimating probability distribution function in stochastic oil reservoir simulation*, Water Resources Research, to appear, 2016. (DOI: http://dx.doi.org/10.1002/2016WR019475)