School of Mathematics, University of Edinburgh

Smoothed circulant embedding and applications in multilevel Monte Carlo methods

MASCOT-NUM 2024

Anastasia Istratuca, **Aretha Teckentrup**

April 4th, 2024

Outline

Motivation

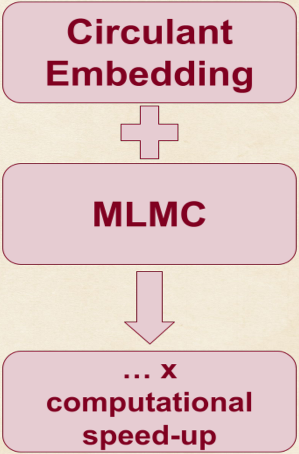
Multilevel Monte Carlo Methods

Circulant Embedding

Smoothing

Numerical Experiments

Conclusions and Outlook



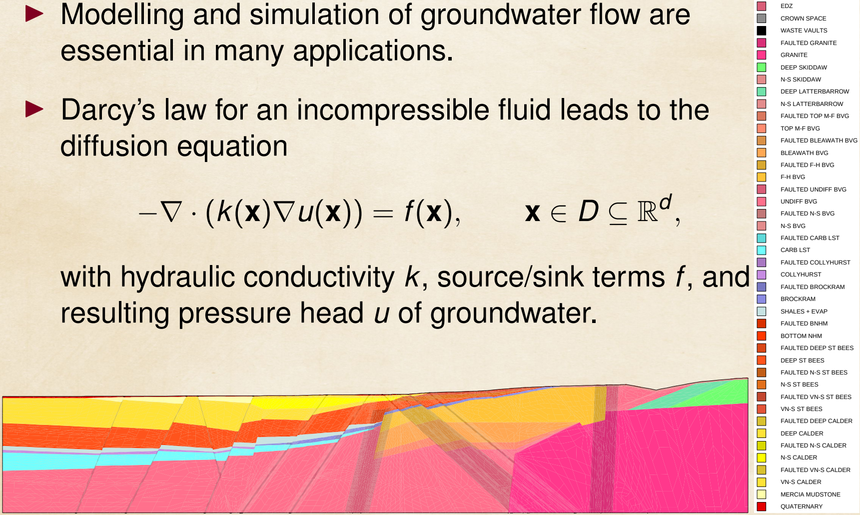
Motivation

Application in Groundwater Flow

- Modelling and simulation of groundwater flow are essential in many applications.
- Darcy's law for an incompressible fluid leads to the diffusion equation

$$-\nabla \cdot (k(\mathbf{x})\nabla u(\mathbf{x})) = f(\mathbf{x}), \qquad \mathbf{x} \in D \subseteq \mathbb{R}^d,$$

with hydraulic conductivity k, source/sink terms f, and resulting pressure head u of groundwater.



Motivation
Application in Groundwater Flow

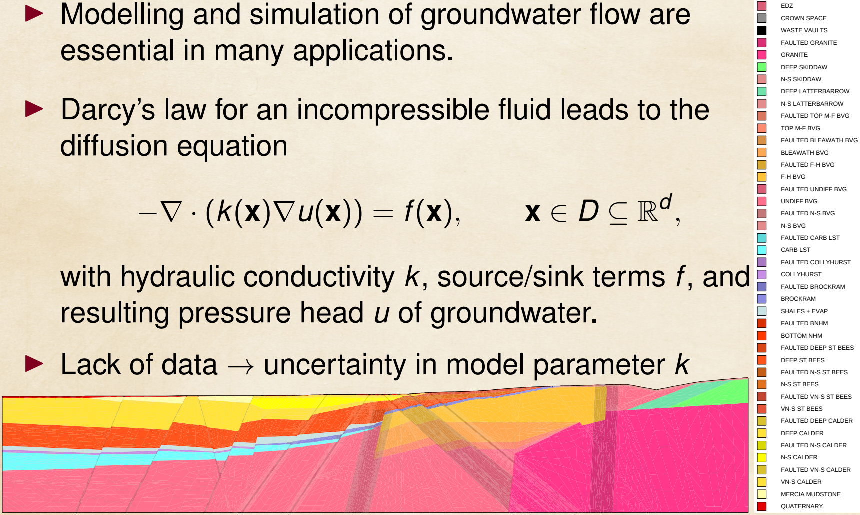
- Modelling and simulation of groundwater flow are essential in many applications.

- Darcy's law for an incompressible fluid leads to the diffusion equation

$$-\nabla \cdot (k(\mathbf{x})\nabla u(\mathbf{x})) = f(\mathbf{x}), \qquad \mathbf{x} \in D \subseteq \mathbb{R}^d,$$

with hydraulic conductivity k, source/sink terms f, and resulting pressure head u of groundwater.

- Lack of data $\rightarrow$ uncertainty in model parameter k



Motivation

Application in Groundwater Flow

- Uncertainty in k propagates through the model, inducing uncertainty in pressure head u.
- We quantify the impact of uncertainty on outputs through *stochastic modelling* ($\rightarrow$ random fields):

$$
\begin{aligned}
-\nabla \cdot (k(\mathbf{x},\omega)\nabla u(\mathbf{x},\omega)) &= f(\mathbf{x}), \quad && \mathbf{x} \in (0,1)^2 \\
u|_{x_1=0} &= 1, && u|_{x_1=1} = 0, \\
\left.\frac{\partial u}{\partial \mathbf{n}}\right|_{x_2=0} &= 0, && \left.\frac{\partial u}{\partial \mathbf{n}}\right|_{x_2=1} = 0.
\end{aligned}
$$

- Uncertainty in k propagates through the model, inducing uncertainty in pressure head u.
- We quantify the impact of uncertainty on outputs through *stochastic modelling* ($\rightarrow$ random fields):

$$\begin{aligned} -\nabla \cdot (k(\mathbf{x},\omega)\nabla u(\mathbf{x},\omega)) &= f(\mathbf{x}), \quad && \mathbf{x} \in (0,1)^2 \\ u|_{x_1=0} &= 1, && u|_{x_1=1} = 0, \\ \frac{\partial u}{\partial \mathbf{n}}\Big|_{x_2=0} &= 0, && \frac{\partial u}{\partial \mathbf{n}}\Big|_{x_2=1} = 0. \end{aligned}$$

- We are usually interested in finding $\mathbb{E}[Q]$, e.g. $Q = u(\mathbf{x}^*, \cdot)$ or Q being the travel time of contaminant particles.

Multilevel Monte Carlo methods
Sampling

Suppose we are interested in finding $\mathbb{E}[Q]$, e.g. $Q = u(\mathbf{x}^*, \cdot)$. Then:

$$-\nabla \cdot (k(\mathbf{x}, \omega^{(i)}) \nabla u(\mathbf{x}, \omega^{(i)})) = f(\mathbf{x})$$

$$\downarrow$$

$$u_h(\mathbf{x}, \omega^{(i)}) \approx u(\mathbf{x}, \omega^{(i)})$$

$$\downarrow$$

$$Q_h^{(i)} \approx Q^{(i)}$$

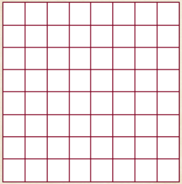
for **one sample** $k\left(\mathbf{x}, \omega^{(i)}\right)$, using e.g. finite elements.

For N i.i.d. samples of $k(\mathbf{x}, \cdot)$:

$$\mathbb{E}[Q_h] \approx \widehat{Q}^{\mathrm{MC}}_{h,N} := \frac{1}{N} \sum_{i=1}^{N} Q_h^{(i)}.$$

Problem: N is typically very large and h is very small:

$$e\left(\widehat{Q}^{\mathrm{MC}}_{h,N}\right)^2 := \mathbb{E}\left[\left(\widehat{Q}^{\mathrm{MC}}_{h,N} - \mathbb{E}[Q]\right)^2\right] = \frac{1}{N}\mathbb{V}[Q_h] + \left(\mathbb{E}[Q_h - Q]\right)^2.$$



Multilevel Monte Carlo Methods

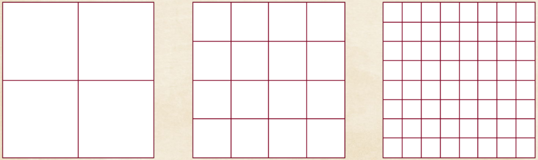
Multilevel Monte Carlo [Heinrich (2001), Giles (2008)]

Solution: spread the approximation cost over multiple "levels":

$$\mathbb{E}[Q_{h_L}] \approx \widehat{Q}_L^{\text{MLMC}} := \mathbb{E}[Q_{h_0}] + \sum_{\ell=1}^{L} \mathbb{E}[Q_{h_\ell} - Q_{h_{\ell-1}}]$$

$$= \frac{1}{N_0} \sum_{i=1}^{N_0} Q_{h_0}^{(i)} + \sum_{\ell=1}^{L} \left(\frac{1}{N_\ell} \sum_{i=1}^{N_\ell} \left(Q_{h_\ell}^{(i)} - Q_{h_{\ell-1}}^{(i)} \right) \right),$$

where $h_\ell = 2^{-\ell} h_0$ and $N_0 > N_1 > \ldots > N_L$.



Multilevel Monte Carlo Methods

Multilevel Monte Carlo [Heinrich (2001), Giles (2008)]

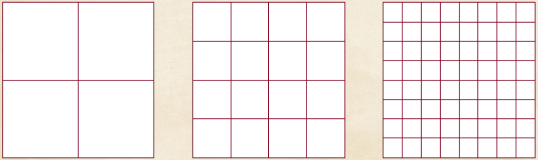
Solution: spread the approximation cost over multiple “levels”:

$$\mathbb{E}[Q_{h_L}] \approx \widehat{Q}_L^{\text{MLMC}} := \mathbb{E}[Q_{h_0}] + \sum_{\ell=1}^{L} \mathbb{E}[Q_{h_\ell} - Q_{h_{\ell-1}}]$$

$$= \frac{1}{N_0}\sum_{i=1}^{N_0} Q_{h_0}^{(i)} + \sum_{\ell=1}^{L}\left(\frac{1}{N_\ell}\sum_{i=1}^{N_\ell}\left(Q_{h_\ell}^{(i)} - Q_{h_{\ell-1}}^{(i)}\right)\right),$$

where $h_\ell = 2^{-\ell} h_0$ and $N_0 > N_1 > \ldots > N_L$. This gives:

$$e\left(\widehat{Q}_L^{\text{MLMC}}\right)^2 := \mathbb{E}\left[\left(\widehat{Q}_L^{\text{MLMC}} - \mathbb{E}[Q]\right)^2\right] = \sum_{\ell=0}^{L} \frac{1}{N_\ell}\mathbb{V}[Y_\ell] + \left(\mathbb{E}[Q_{h_L} - Q]\right)^2$$



Theorem (Complexity of Multilevel Monte Carlo)

Assume that

(A1) $|\mathbb{E}[Q_h] - \mathbb{E}[Q]| \leq C_1\, h^{\alpha}$ *(bias decay)*

(A2) $\mathbb{V}[Q_{h_\ell} - Q_{h_{\ell-1}}] \leq C_2\, h_\ell^{\beta}$ *(variance decay)*

(A3) $\mathrm{Cost}(Q_h^{(i)}) \leq C_3\, h^{-\gamma}$ *(cost of one sample)*

for some constants $C_1, C_2, C_3, \alpha, \beta, \gamma > 0$ *with* $2\alpha \geq \min(\beta, \gamma)$.

Then there exist L *and* $\{N_\ell\}_{\ell=0}^{L}$ *such that* $e\left(\widehat{Q}_L^{MLMC}\right)^2 \leq \varepsilon^2$ *and*

$$\mathrm{Cost}(\widehat{Q}_L^{MLMC}) = \begin{cases} \mathcal{O}(\varepsilon^{-2}) & \text{if } \beta > \gamma, \\ \mathcal{O}(\varepsilon^{-2} \log(\varepsilon)^2) & \text{if } \beta = \gamma, \\ \mathcal{O}(\varepsilon^{-2-(\gamma-\beta)/\alpha}) & \text{if } \beta < \gamma. \end{cases}$$

There are three different cases in the complexity theorem:

- $\beta > \gamma$: the majority of computational cost is on level 0. In this case $\mathbb{V}[Q_{h_\ell} - Q_{h_{\ell-1}}]$, and hence N_ℓ, decays quickly with ℓ and we do a negligible number of samples on level L.
- $\beta = \gamma$: the computational cost is spread evenly across the levels.
- $\beta < \gamma$: the majority of computational cost is on level L. In this case $\mathrm{Cost}(Q_{h_\ell}^{(i)})$ grows very quickly with ℓ and just one sample on level L adds significantly to the cost.

Multilevel Monte Carlo Methods
Complexity III [Giles (2008), Cliffe et al (2011)]

With an optimal linear solver (i.e. $\gamma \approx d$), and standard piece-wise linear finite elements (i.e. $\alpha = 1$ and $\beta = 2$), the computational ε-costs for the Darcy problem are bounded by:

d	**MLMC**	**MC**
1	$\mathcal{O}(\varepsilon^{-2})$	$\mathcal{O}(\varepsilon^{-3})$
2	$\mathcal{O}(\varepsilon^{-2})$	$\mathcal{O}(\varepsilon^{-4})$
3	$\mathcal{O}(\varepsilon^{-3})$	$\mathcal{O}(\varepsilon^{-5})$

For $\varepsilon = 10^{-3}$ and $d = 3$, the costs of MLMC and MC are $\mathcal{O}(10^9)$ and $\mathcal{O}(10^{15})$, respectively.

Suppose $k(\mathbf{x}, \cdot)$ is a log-normal random field, so that:

$$k(\mathbf{x}, \omega) = \exp(Z(\mathbf{x}, \omega)),$$

where $Z(\mathbf{x}, \cdot)$ is a Gaussian random field with:

$$\begin{aligned} &\mathbb{E}[Z(\mathbf{x}, \cdot)] \equiv 0 \\ &\mathbb{E}[Z(\mathbf{x}, \cdot)Z(\mathbf{y}, \cdot)] = r(\mathbf{x}, \mathbf{y}) = C(\mathbf{x} - \mathbf{y}). \end{aligned} \tag{1}$$

Suppose $k(\mathbf{x}, \cdot)$ is a log-normal random field, so that:

$$k(\mathbf{x}, \omega) = \exp(Z(\mathbf{x}, \omega)),$$

where $Z(\mathbf{x}, \cdot)$ is a Gaussian random field with:

$$\begin{aligned} \mathbb{E}[Z(\mathbf{x}, \cdot)] &\equiv 0 \\ \mathbb{E}[Z(\mathbf{x}, \cdot) Z(\mathbf{y}, \cdot)] &= r(\mathbf{x}, \mathbf{y}) = C(\mathbf{x} - \mathbf{y}). \end{aligned} \tag{1}$$

The covariance function C selected for this application is given by (Hoeksema and Kitanidis, 1985):

$$C(\mathbf{r}) := \sigma^2 \exp\left(-\frac{\|\mathbf{r}\|_1}{\rho} \right).$$

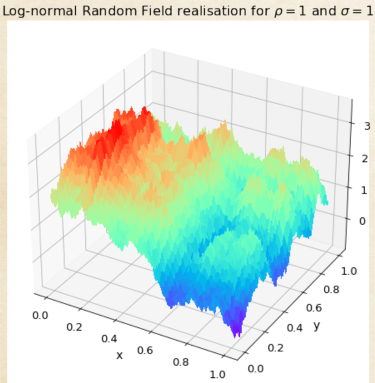


Figure: $\rho = 1, \sigma = 1$

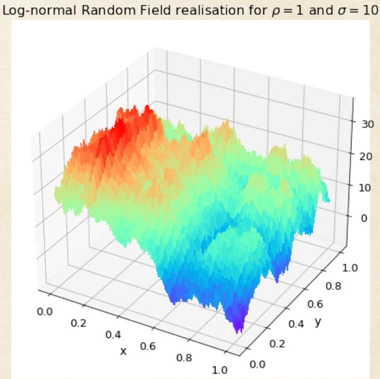


Figure: $\rho = 1, \sigma = 10$

Circulant Embedding

Random Fields - Example II

Log-normal Random Field realisation for $\rho = 1$ and $\sigma = 1$

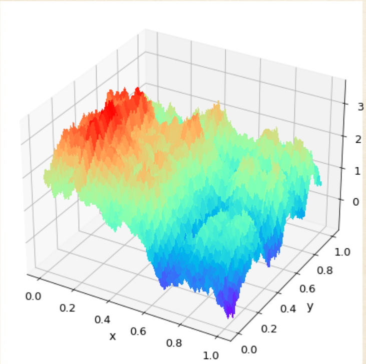


Figure: $\rho = 1, \sigma = 1$

Log-normal Random Field realisation for $\rho = 0.1$ and $\sigma = 1$

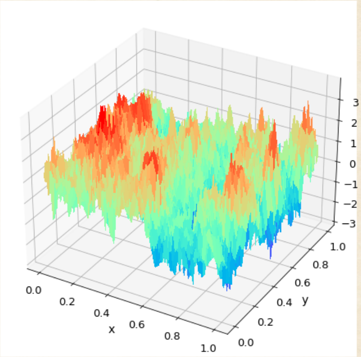


Figure: $\rho = 0.1, \sigma = 1$

How do we obtain samples of $k(\mathbf{x}, \omega)$ on mesh $\mathcal{T}$?

How do we obtain samples of $k(\mathbf{x}, \omega)$ on mesh $\mathcal{T}$?

For any factorisation of the covariance matrix R of $Z_{\mathcal{T}}(\mathbf{x}, \cdot)$:

$$R = \Theta\Theta^T,$$

and any vector ξ such that:

$$\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, I),$$

we can take $\mathbf{Z} := \Theta\xi$ to obtain $\mathbf{Z} \sim Z_{\mathcal{T}}(\mathbf{x}, \cdot)$.

Challenge: many classical factorisation methods, such as Cholesky, have cubic cost in the number of mesh points!

How do we obtain samples $k(\mathbf{x}, \omega)$ on mesh $\mathcal{T}$?

How do we obtain samples $k(\mathbf{x}, \omega)$ on mesh $\mathcal{T}$?

$\mathcal{T}$

where:

- $\mathcal{T}$ - uniform two-dimensional discretisation mesh;
- R - covariance matrix;
- S - circulant embedding matrix;
- $G = \Re(F) + \Im(F)$, F - two-dimensional Fourier matrix;
- $\Lambda = \sqrt{4 m_1 m_2} F \mathbf{s}$ - diagonal matrix of eigenvalues of S, with $\mathbf{s}$ the first column of S;
- $\mathbf{Z}$ - sample from the Gaussian field.

How do we obtain samples $k(\mathbf{x}, \omega)$ on mesh $\mathcal{T}$?

$$\mathcal{T} \xrightarrow{C(\mathbf{x}_i - \mathbf{y}_j)} R$$

where:

- $\mathcal{T}$ - uniform two-dimensional discretisation mesh;
- R - covariance matrix;
- S - circulant embedding matrix;
- $G = \Re(F) + \Im(F)$, F - two-dimensional Fourier matrix;
- $\Lambda = \sqrt{4 m_1 m_2} F\mathbf{s}$ - diagonal matrix of eigenvalues of S, with $\mathbf{s}$ the first column of S;
- $\mathbf{Z}$ - sample from the Gaussian field.

Circulant Embedding

Overview [Dietrich and Newsam (1993)]

How do we obtain samples $k(\mathbf{x}, \omega)$ on mesh $\mathcal{T}$?

$$\mathcal{T} \xrightarrow{C(\mathbf{x}_i - \mathbf{y}_j)} R \xrightarrow{\text{embedding}} S$$

where:

- $\mathcal{T}$ - uniform two-dimensional discretisation mesh;
- R - covariance matrix;
- S - circulant embedding matrix;
- $G = \Re(F) + \Im(F)$, F - two-dimensional Fourier matrix;
- $\Lambda = \sqrt{4 m_1 m_2} F \mathbf{s}$ - diagonal matrix of eigenvalues of S, with $\mathbf{s}$ the first column of S;
- $\mathbf{Z}$ - sample from the Gaussian field.

Circulant Embedding

Overview [Dietrich and Newsam (1993)]

How do we obtain samples $k(\mathbf{x}, \omega)$ on mesh $\mathcal{T}$?

$$\mathcal{T} \xrightarrow{C(\mathbf{x}_i - \mathbf{y}_j)} R \xrightarrow{\text{embedding}} S \xrightarrow[\text{transform}]{\text{Fourier}} S = G\Lambda G^T$$

where:

- $\mathcal{T}$ - uniform two-dimensional discretisation mesh;
- R - covariance matrix;
- S - circulant embedding matrix;
- $G = \Re(F) + \Im(F)$, F - two-dimensional Fourier matrix;
- $\Lambda = \sqrt{4 m_1 m_2} F\mathbf{s}$ - diagonal matrix of eigenvalues of S, with $\mathbf{s}$ the first column of S;
- $\mathbf{Z}$ - sample from the Gaussian field.

Circulant Embedding

Overview [Dietrich and Newsam (1993)]

How do we obtain samples $k(\mathbf{x}, \omega)$ on mesh $\mathcal{T}$?

$$\mathcal{T} \xrightarrow{C(\mathbf{x}_i - \mathbf{y}_j)} R \xrightarrow{\text{embedding}} S \xrightarrow[\text{transform}]{\text{Fourier}} S = G \Lambda G^T \xrightarrow{\xi \sim \mathcal{N}(\mathbf{0}, I)} \mathbf{Z} = G \Lambda^{1/2} \xi,$$

where:

- $\mathcal{T}$ - uniform two-dimensional discretisation mesh;
- R - covariance matrix;
- S - circulant embedding matrix;
- $G = \Re(F) + \Im(F)$, F - two-dimensional Fourier matrix;
- $\Lambda = \sqrt{4 m_1 m_2} F \mathbf{s}$ - diagonal matrix of eigenvalues of S, with $\mathbf{s}$ the first column of S;
- $\mathbf{Z}$ - sample from the Gaussian field.

Circulant Embedding

How? (1d for simplicity)

$$R = \begin{bmatrix} C_0 & C_1 & \dots & C_m \\ C_1 & C_0 & \dots & C_{m-1} \\ \vdots & \vdots & \ddots & \vdots \\ C_m & C_{m-1} & \dots & C_0 \end{bmatrix}, \quad C_i = C\left(\frac{i}{m}\right)$$

Circulant Embedding

How? (1d for simplicity)

$$R = \begin{bmatrix} C_0 & C_1 & \dots & C_m \\ C_1 & C_0 & \dots & C_{m-1} \\ \vdots & \vdots & \ddots & \vdots \\ C_m & C_{m-1} & \dots & C_0 \end{bmatrix}, \quad C_i = C\left(\frac{i}{m}\right)$$

$$\downarrow$$

$$S = \begin{bmatrix} C_0 & C_1 & \dots & C_m & & \\ C_1 & C_0 & \dots & C_{m-1} & & \\ \vdots & \vdots & \ddots & \vdots & & \\ C_m & C_{m-1} & \dots & C_0 & & \\ & & & & & \\ & & & & & \end{bmatrix}$$

Circulant Embedding

How? (1d for simplicity)

$$R = \begin{bmatrix} C_0 & C_1 & \dots & C_m \\ C_1 & C_0 & \dots & C_{m-1} \\ \vdots & \vdots & \ddots & \vdots \\ C_m & C_{m-1} & \dots & C_0 \end{bmatrix}, \quad C_i = C\left(\frac{i}{m}\right)$$

$$\downarrow$$

$$S = \begin{bmatrix} C_0 & C_1 & \dots & C_m & C_{m-1} & \dots & C_1 \\ C_1 & C_0 & \dots & C_{m-1} & C_m & \dots & C_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ C_m & C_{m-1} & \dots & C_0 & C_1 & \dots & C_{m-1} \\ & & & & & & \\ & & & & & & \end{bmatrix}$$

Circulant Embedding

How? (1d for simplicity)

$$R = \begin{bmatrix} C_0 & C_1 & \dots & C_m \\ C_1 & C_0 & \dots & C_{m-1} \\ \vdots & \vdots & \ddots & \vdots \\ C_m & C_{m-1} & \dots & C_0 \end{bmatrix}, \qquad C_i = C\left(\frac{i}{m}\right)$$

$$\downarrow$$

$$S = \begin{bmatrix} C_0 & C_1 & \dots & C_m & C_{m-1} & \dots & C_1 \\ C_1 & C_0 & \dots & C_{m-1} & C_m & \dots & C_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ C_m & C_{m-1} & \dots & C_0 & C_1 & \dots & C_{m-1} \\ C_{m-1} & C_{m-2} & \dots & C_1 & & & \\ \vdots & \vdots & \ddots & \vdots & & & \\ C_1 & C_2 & \dots & C_{m-1} & & & \end{bmatrix}$$

Circulant Embedding
How? (1d for simplicity)

$$R = \begin{bmatrix} c_0 & c_1 & \dots & c_m \\ c_1 & c_0 & \dots & c_{m-1} \\ \vdots & \vdots & \ddots & \vdots \\ c_m & c_{m-1} & \dots & c_0 \end{bmatrix}, \quad c_i = C\left(\frac{i}{m}\right)$$

$$\downarrow$$

$$S = \begin{bmatrix} c_0 & c_1 & \dots & c_m & c_{m-1} & \dots & c_1 \\ c_1 & c_0 & \dots & c_{m-1} & c_m & \dots & c_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ c_m & c_{m-1} & \dots & c_0 & c_1 & \dots & c_{m-1} \\ c_{m-1} & c_{m-2} & \dots & c_1 & c_0 & \dots & c_m \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ c_1 & c_2 & \dots & c_{m-1} & c_m & \dots & c_0 \end{bmatrix}$$

Issue: If the random field is extremely oscillatory (small ρ), these fluctuations cannot be resolved on a very coarse grid.

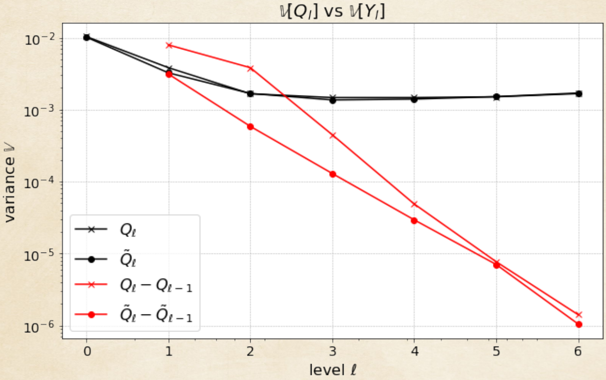


Figure: $Q = u(\mathbf{x}^*)$, $\nu = 1.5$ and $\rho = 0.03$.

Heuristically: $h_0 \leq \rho$ (or $h_0 \leq \sqrt{8\nu}\rho$ for Matérn kernels).

Solution: "Smooth" samples of $k(\mathbf{x}, \cdot)$ so that bulk behaviour is captured correctly, and variations are resolved more easily.

Smoothing

Overview [Istratuca, T. (submitted 2023)]

Solution: "Smooth" samples of $k(\mathbf{x}, \cdot)$ so that bulk behaviour is captured correctly, and variations are resolved more easily.

How: Drop the τ smallest eigenvalues in a given sample $\mathbf{Z} = G\Lambda^{1/2}\xi$, which correspond to the sharpest oscillations.

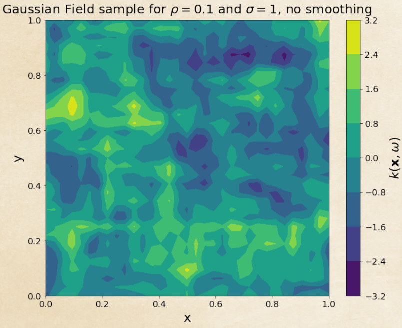


Figure: Without smoothing

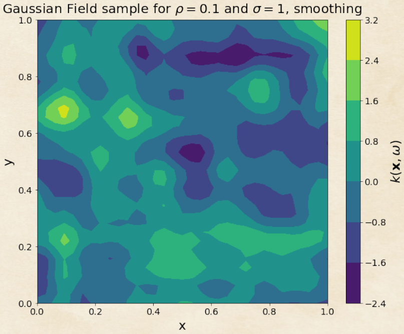


Figure: With smoothing

Smoothing
Example

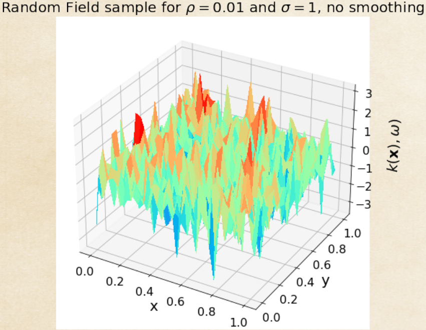


Figure: Without smoothing

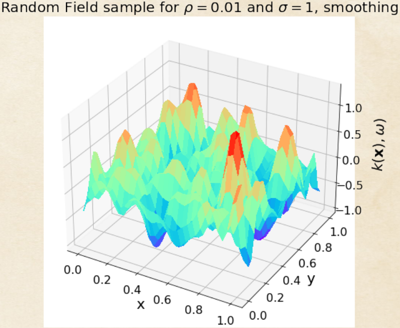


Figure: With smoothing

Smoothing

Error [Istratuca, T. (submitted 2023)]

Theorem

Let τ be the truncation index and $\tilde{\mathbf{Z}}$ be the resulting smoothed sample. Then, for any $p \in \mathbb{N}$:

$$\mathbb{E}\left[\|\mathbf{Z} - \tilde{\mathbf{Z}}\|_\infty^p\right] \lesssim s^{-\frac{p}{2}} \left(\max_{j=s-\tau+1,\ldots,s} \sqrt{\lambda_j}\right)^p \tau^p,$$

where $s = \prod_{i=1}^d 2(m_i + J_i)$.

- Here, s is the dimension of the circulant matrix S.
- We have m_i mesh points in $\mathcal{T}$ in dimension i.
- J_i are "padding" values that might be necessary to ensure S is symmetric positive definite. (Not needed for the covariance function considered in this talk.)

The preceding theorem can be used to obtain convergence rates in τ of $Q_h - \tilde{Q}_h$, given convergence rates of the eigenvalues.

Theorem

Let τ be the truncation index and $\tilde{Q}_h$ be the resulting smoothed quantity of interest. Then, for C as in (1) and any $p \in [1, \infty)$:

$$\mathbb{E}\left[|Q_h - \tilde{Q}_h|^p\right] \lesssim (s - \tau + 1)^{-p}\, \tau^p.$$

where $s = \prod_{i=1}^{d} 2m$ and $m = h^{-1} + 1$.

Similar results are obtained for Matérn covariance kernels.

- Using smoothing in multilevel Monte Carlo, we introduce a level-dependent truncation index τ_ℓ.
- We choose $\tau_L = 0$, so that this strategy does not introduce additional bias in the final result.

Smoothing

Multilevel Monte Carlo [Istratuca, T. (submitted 2023)]

- Using smoothing in multilevel Monte Carlo, we introduce a level-dependent truncation index τ_ℓ.
- We choose $\tau_L = 0$, so that this strategy does not introduce additional bias in the final result.
- Combining the preceding theorem with an error bound on $Q - Q_h$, i.e. the finite element error, we obtain

$$\mathbb{E}[|Q - \tilde{Q}_{h_\ell}|] \leq C\, h_\ell^\alpha + C'\,(s_\ell - \tau_\ell + 1)^{-1}\, \tau_\ell.$$

- We choose τ_ℓ as a function of h_ℓ to balance the two error contributions.
- Note that this means that the convergence rates α and β in the multilevel Monte Carlo complexity theorem are unchanged.

Numerical Experiments

Computational complexity - $\rho = 0.3$ for $Q = u(\mathbf{x}^*)$

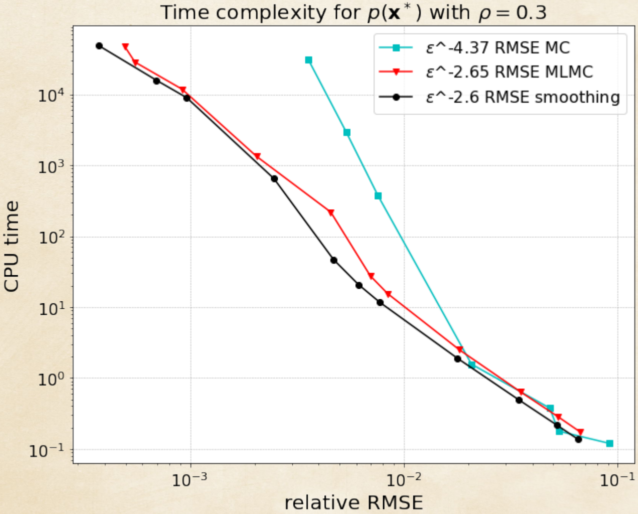


Figure: MC vs MLMC vs MLMC with smoothing for $\rho = 0.3$.

Numerical Experiments

Computational complexity - $\rho = 0.1$ for $Q = u(\mathbf{x}^*)$

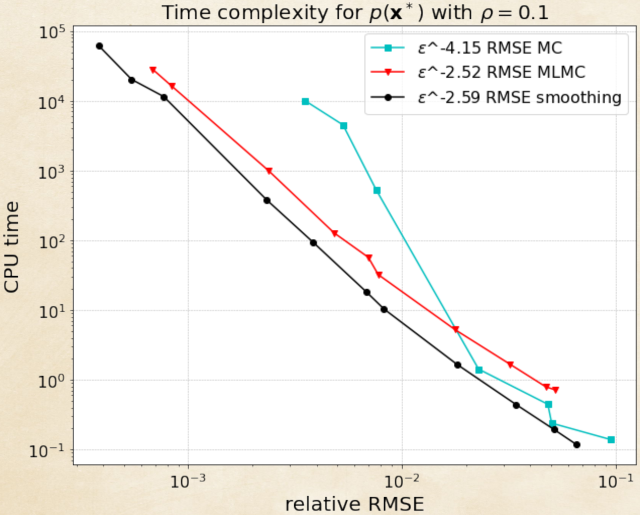


Figure: MC vs MLMC vs MLMC with smoothing for $\rho = 0.1$.

Conclusions and Outlook

- Circulant Embedding is an efficient technique for sampling exactly from a random field on a discrete mesh.
- Introducing smoothing improves the performance of multilevel Monte Carlo in the case of small correlation lengths in the random hydraulic conductivity.

Conclusions and Outlook

- Circulant Embedding is an efficient technique for sampling exactly from a random field on a discrete mesh.
- Introducing smoothing improves the performance of multilevel Monte Carlo in the case of small correlation lengths in the random hydraulic conductivity.
- Similar ideas can be used in other sampling methods, e.g. Karhunen-Loève expansions [Ullmann et al 2013].
- Smoothing can be introduced in multilevel Markov chain Monte Carlo, see e.g. [Dodwell et al 2019] and references therein.
- Multilevel Monte Carlo can do more than compute expected values - see Mike Giles' community website!

References I

[1] K. A. Cliffe et al. "Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients". In: *Computing and Visualization in Science* 14.1 (2011), p. 3.

[2] C. R. Dietrich and G. N. Newsam. "A fast and exact method for multidimensional Gaussian stochastic simulations". en. In: *Water Resources Research* 29 (1993), pp. 2861–2869.

[3] T. Dodwell et al. "A hierarchical multilevel Markov chain Monte Carlo algorithm with applications to uncertainty quantification in subsurface flow". In: *SIAM/ASA Journal on Uncertainty Quantification* 3.1 (2015), pp. 1075–1108.

[4] M. B. Giles. "Multilevel Monte Carlo methods". In: *Acta Numerica* 24 (2015), pp. 259–328.

[5] M. B. Giles. "Multilevel Monte Carlo Path Simulation". In: *Operations Research* 56.3 (2008), pp. 607–617.

[6] S Heinrich. "Multilevel Monte Carlo methods". In: *Large-Scale Scientific Computing: Third International Conference*. Springer, 2001, pp. 58–67.

References II

[7] R. J. Hoeksema and P. K. Kitanidis. "Analysis of the Spatial Structure of Properties of Selected Aquifers". In: *Water Resources Research* 21 (1985), pp. 563–572.

[8] A. Istratuca and A. L. Teckentrup. *Smoothed Circulant Embedding with Applications to Multilevel Monte Carlo Methods for PDEs with Random Coefficients*. 2023. arXiv: 2306.13493.

Thank you! Questions?