An Overview of Multilevel Monte Carlo Techniques for Solving PDEs with Random Coefficients

Pieterjan Robbe
Dirk Nuyens
Stefan Vandewalle

Multilevel and Multigrid Workshop, TU Delft

KU LEUVEN

Department of Computer Science
NUMA research group
Celestijnenlaan 200A bus 2402
3001 Leuven, Belgium

euforia

Contents

PART 1

Classic MLMC

The KU Leuven UQ team

- **NUMA**: numerical analysis and applied mathematics
- 11 professors, about 40 postdocs and PhD students
- Working on UQ:

S. Vandewalle
Professor

P. Robbe
PhD student

A. Van Barel
PhD student

P. Blondeel
PhD student

+ collaborations with prof. D. Nuyens and prof. G. Samaey

Model parametric elliptic PDE

$$-\nabla \cdot a(\boldsymbol{x}, \boldsymbol{y}) \nabla u(\boldsymbol{x}, \boldsymbol{y}) = f(\boldsymbol{x})$$

Model parametric elliptic PDE

$$-\nabla \cdot a(\boldsymbol{x}, \boldsymbol{y}) \nabla u(\boldsymbol{x}, \boldsymbol{y}) = f(\boldsymbol{x})$$

- early work
  - [Ghanem, Spanos, 1997]
  - [Babuska, Tempone, Zouraris, 2004]
  - [Babuska, Nobile, Tempone, 2007]
  - and many others
- parametric PDE setting in
  - [Cohen, DeVore, Schwab, 2011]
- recent interest from multilevel/QMC community
  - [Graham, Kuo, Nuyens, Scheichl, Sloan, 2011]
  - [Cliffe, Giles, Scheichl, Teckentrup, 2011]
  - [Kuo, Schwab, Sloan, 2012]
  - [Kuo, Nuyens, 2016]
  - and many others

Example

- $a(\boldsymbol{x}, \boldsymbol{y})$ is derived from a **Gaussian random field** $z(\boldsymbol{x}, \boldsymbol{y})$ with given mean $z_0(\boldsymbol{x})$ and covariance function, e.g.,

$$C(\boldsymbol{x}, \boldsymbol{x}') = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu} r\right)^{\nu} K_{\nu} \left(\sqrt{2\nu} r\right), \quad r = \frac{\|\boldsymbol{x} - \boldsymbol{x}'\|_2}{\lambda_c}$$

Example

- $a(\boldsymbol{x}, \boldsymbol{y})$ is derived from a **Gaussian random field** $z(\boldsymbol{x}, \boldsymbol{y})$ with given mean $z_0(\boldsymbol{x})$ and covariance function, e.g.,

$$C(\mathbf{x}, \mathbf{x}') = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu} r\right)^{\nu} K_{\nu} \left(\sqrt{2\nu} r\right), \quad r = \frac{\|\mathbf{x} - \mathbf{x}'\|_2}{\lambda_c}$$

- Samples can be generated using a **KL expansion**

$$z(\boldsymbol{x}, \boldsymbol{y}) = \sum_{j \geq 1} y_j \sqrt{\theta_j} \psi_j(\boldsymbol{x})$$

where the eigenvalues θ_j and eigenfunctions $\psi_j(\boldsymbol{x})$ satisfy

$$\int_D C(\mathbf{x}, \mathbf{x}') \psi_j(\mathbf{x}') \mathrm{d}\mathbf{x}' = \theta_j \psi_j(\boldsymbol{x})$$

and the y_j are standard normal random numbers

Example

- $a(\boldsymbol{x}, \boldsymbol{y})$ is derived from a **Gaussian random field** $z(\boldsymbol{x}, \boldsymbol{y})$ with given mean $z_0(\boldsymbol{x})$ and covariance function, e.g.,

$$C(\boldsymbol{x}, \boldsymbol{x}') = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu} r\right)^{\nu} K_{\nu} \left(\sqrt{2\nu} r\right), \quad r = \frac{\|\boldsymbol{x} - \boldsymbol{x}'\|_2}{\lambda_c}$$

- Samples can be generated using a **KL expansion**

$$z(\boldsymbol{x}, \boldsymbol{y}) = \sum_{j \geq 1} y_j \sqrt{\theta_j} \psi_j(\boldsymbol{x})$$

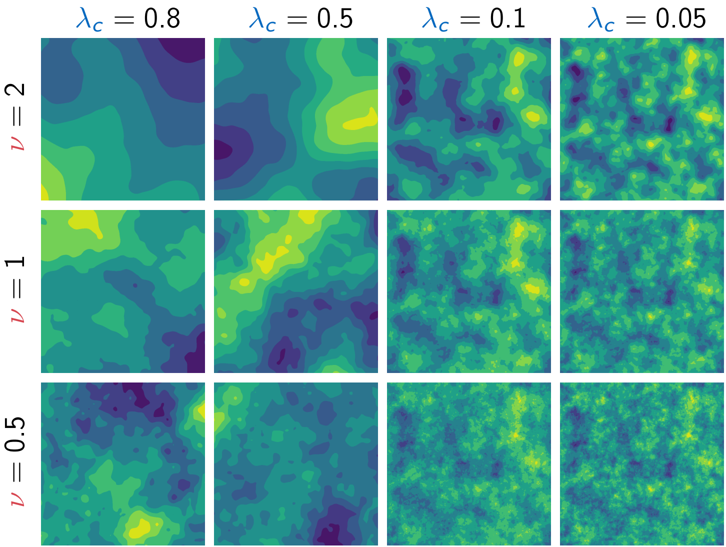
where the eigenvalues θ_j and eigenfunctions $\psi_j(\boldsymbol{x})$ satisfy

$$\int_D C(\boldsymbol{x}, \boldsymbol{x}') \psi_j(\boldsymbol{x}') \mathrm{d}\boldsymbol{x}' = \theta_j \psi_j(\boldsymbol{x})$$

and the y_j are standard normal random numbers

- $a(\boldsymbol{x}, \boldsymbol{y}) = \exp(z(\boldsymbol{x}, \boldsymbol{y}))$ is known as the "*lognormal case*"

Example Gaussian random fields



see GaussianRandomFields.jl

Sampling based methods

- Goal: compute statistics of **quantity of interest**

$$Q = F(u(\boldsymbol{x}, \boldsymbol{y}))$$

quantity of interest is uncertain, denote $F(\boldsymbol{y}) := F(u(\boldsymbol{x}, \boldsymbol{y}))$

Sampling based methods

- Goal: compute statistics of **quantity of interest**

$$Q = F(u(\boldsymbol{x}, \boldsymbol{y}))$$

quantity of interest is uncertain, denote $F(\boldsymbol{y}) := F(u(\boldsymbol{x}, \boldsymbol{y}))$

```
Draw a sufficiently large sample set y^(1), y^(2), ..., y^(N)
for n = 1 to N do
    Compute the random field a(x, y^(n))
    Solve a deterministic PDE using method of choice
    Compute the quantity of interest F(y^(n))
end for
```

Sampling based methods

- Goal: compute statistics of **quantity of interest**

$$Q = F(u(\boldsymbol{x}, \boldsymbol{y}))$$

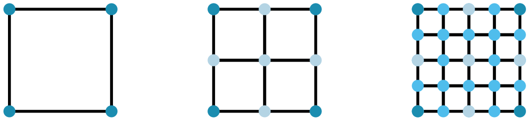
quantity of interest is uncertain, denote $F(\boldsymbol{y}) := F(u(\boldsymbol{x}, \boldsymbol{y}))$

```
Draw a sufficiently large sample set y^(1), y^(2), ..., y^(N)
for n = 1 to N do
    Compute the random field a(x, y^(n))
    Solve a deterministic PDE using method of choice
    Compute the quantity of interest F(y^(n))
end for
```

- Example: the Monte Carlo (MC) estimator for E$[Q]$ is

$$\mathcal{Q}^{\mathrm{MC}} = \frac{1}{N} \sum_{n=1}^{N} F(\boldsymbol{y}^{(n)})$$

A hierarchy of coarser grids



$\ell = 0$ $\ell = 1$ $\ell = 2$

- Solution of the PDE (and hence quantity of interest F) is approximated numerically
- Suppose we have a **hierarchy of approximations** F_ℓ, $\ell = 0, \ldots, L$ and $F_\ell \to F$ as $\ell \to \infty$
- Do not compute $\mathrm{E}[F_L]$ by sampling from F_L, but by sampling from the whole hierarchy F_ℓ, $\ell = 0, \ldots, L$

Multilevel Monte Carlo (MLMC)

- Basis is the **telescoping sum**

$$\mathrm{E}[F_L] = \mathrm{E}[F_0] + \sum_{\ell=1}^{L} \mathrm{E}[F_\ell - F_{\ell-1}]$$

Multilevel Monte Carlo (MLMC)

- Basis is the **telescoping sum**

$$\mathrm{E}[F_L] = \mathrm{E}[F_0] + \sum_{\ell=1}^{L} \mathrm{E}[F_\ell - F_{\ell-1}]$$

- MLMC estimator uses Monte Carlo to estimate each term:

$$\mathcal{Q}_L^{\mathrm{MC}} = \sum_{\ell=0}^{L} \frac{1}{N_\ell} \sum_{n=1}^{N_\ell} \left(F_\ell(\boldsymbol{y}_\ell^{(n)}) - F_{\ell-1}(\boldsymbol{y}_\ell^{(n)}) \right) \quad (F_{-1} := 0)$$

Multilevel Monte Carlo (MLMC)

- Basis is the **telescoping sum**

$$\mathrm{E}[F_L] = \mathrm{E}[F_0] + \sum_{\ell=1}^{L} \mathrm{E}[F_\ell - F_{\ell-1}]$$

- MLMC estimator uses Monte Carlo to estimate each term:

$$\mathcal{Q}_L^{\mathrm{MC}} = \sum_{\ell=0}^{L} \frac{1}{N_\ell} \sum_{n=1}^{N_\ell} \left(F_\ell(\boldsymbol{y}_\ell^{(n)}) - F_{\ell-1}(\boldsymbol{y}_\ell^{(n)}) \right) \quad (F_{-1} := 0)$$

- Crucially, use the **same random numbers** $\boldsymbol{y}_\ell^{(n)}$ in each sample

$$V_\ell := \mathrm{V}[F_\ell - F_{\ell-1}] = \mathrm{V}[F_\ell] + \mathrm{V}[F_{\ell-1}] - 2\mathrm{cov}(F_\ell, F_{\ell-1})$$
$$\ll \mathrm{V}[F_\ell] + \mathrm{V}[F_{\ell-1}]$$

Multilevel Monte Carlo (MLMC)

- Basis is the **telescoping sum**

$$\mathsf{E}[F_L] = \mathsf{E}[F_0] + \sum_{\ell=1}^{L} \mathsf{E}[F_\ell - F_{\ell-1}]$$

- MLMC estimator uses Monte Carlo to estimate each term:

$$\mathcal{Q}_L^{\mathrm{MC}} = \sum_{\ell=0}^{L} \frac{1}{N_\ell} \sum_{n=1}^{N_\ell} \left(F_\ell(\boldsymbol{y}_\ell^{(n)}) - F_{\ell-1}(\boldsymbol{y}_\ell^{(n)}) \right) \quad (F_{-1} := 0)$$

- Crucially, use the **same random numbers** $\boldsymbol{y}_\ell^{(n)}$ in each sample

$$V_\ell := \mathsf{V}[F_\ell - F_{\ell-1}] = \mathsf{V}[F_\ell] + \mathsf{V}[F_{\ell-1}] - 2\mathrm{cov}(F_\ell, F_{\ell-1})$$
$$\ll \mathsf{V}[F_\ell] + \mathsf{V}[F_{\ell-1}]$$

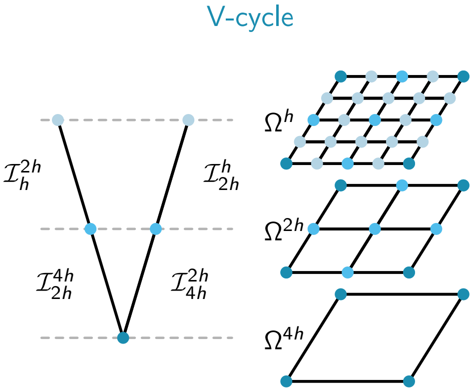
- Most samples will be taken on the coarse grid, where samples are cheap, and only few samples are needed on the finest grid

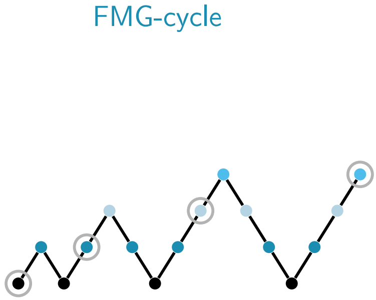
PART 2

Extensions of MLMC

Full Multigrid (FMG)

- **Full Multigrid** can compute a solution to discretization accuracy in $\mathcal{O}(M)$ time, where M is the number of DOF
- FMG also computes free solutions on coarser grids





Multigrid Multilevel Monte Carlo (MG-MLMC)

- Idea is to **recycle** the coarse solutions from the FMG method as coarse samples in the MLMC method
- MG-MLMC estimator [Kumar, Oosterlee, Dwight, 2017]

$$\mathcal{Q}_{L,\text{reuse}}^{\text{MC}} := \sum_{\ell=0}^{L} \left(\frac{1}{\sum_{i=\ell}^{L} N_i} \right) \sum_{k=\ell}^{L} \sum_{n=1}^{N_k} \left(F_\ell(\boldsymbol{y}_k^{(n)}) - F_{\ell-1}(\boldsymbol{y}_k^{(n)}) \right)$$

Multigrid Multilevel Monte Carlo (MG-MLMC)

- Idea is to **recycle** the coarse solutions from the FMG method as coarse samples in the MLMC method
- MG-MLMC estimator [Kumar, Oosterlee, Dwight, 2017]

$$\mathcal{Q}^{\mathrm{MC}}_{L,\mathrm{reuse}} := \sum_{\ell=0}^{L} \left(\frac{1}{\sum_{i=\ell}^{L} N_i} \right) \sum_{k=\ell}^{L} \sum_{n=1}^{N_k} \left(F_\ell(\boldsymbol{y}_k^{(n)}) - F_{\ell-1}(\boldsymbol{y}_k^{(n)}) \right)$$

- This is a sum of $L+1$ estimators $\mathcal{Y}_\ell$, i.e.,

$$\mathcal{Q}^{\mathrm{MC}}_{L,\mathrm{reuse}} := \sum_{\ell=0}^{L} \mathcal{Y}_\ell,$$

  that are **not independent**
- Accuracy of estimator is controlled using mean-square-error

$$\mathrm{MSE}\left(\mathcal{Q}^{\mathrm{MC}}_{L,\mathrm{reuse}}\right) = \mathrm{V}[\mathcal{Q}^{\mathrm{MC}}_{L,\mathrm{reuse}}] + \mathrm{Bias}\left(\mathcal{Q}^{\mathrm{MC}}_{L,\mathrm{reuse}}\right)^2$$

Obtaining a variance estimate

- Variance of the MG-MLMC estimator is

$$\mathsf{V}[\mathcal{Q}_{L,\text{reuse}}^{\text{MC}}] = \sum_{\ell=0}^{L} \mathsf{V}[\mathcal{Y}_\ell] + 2 \sum_{0 \le \ell < \tau \le L} \text{cov}(\mathcal{Y}_\ell, \mathcal{Y}_\tau)$$

$$= \sum_{\ell=0}^{L} \left(\frac{V_\ell}{\sum_{i=\ell}^{L} N_i} \right) + 2 \sum_{0 \le \ell < \tau \le L} \rho_{\ell\tau} \sqrt{\left(\frac{V_\ell}{\sum_{i=\ell}^{L} N_i} \right) \left(\frac{V_\tau}{\sum_{i=\tau}^{L} N_i} \right)}$$

Obtaining a variance estimate

- Variance of the MG-MLMC estimator is

$$\mathrm{V}[\mathcal{Q}_{L,\mathrm{reuse}}^{\mathrm{MC}}] = \sum_{\ell=0}^{L} \mathrm{V}[\mathcal{Y}_\ell] + 2 \sum_{0\le\ell<\tau\le L} \mathrm{cov}(\mathcal{Y}_\ell, \mathcal{Y}_\tau)$$

$$= \sum_{\ell=0}^{L} \left(\frac{V_\ell}{\sum_{i=\ell}^{L} N_i} \right) + 2 \sum_{0\le\ell<\tau\le L} \rho_{\ell\tau} \sqrt{\left(\frac{V_\ell}{\sum_{i=\ell}^{L} N_i} \right) \left(\frac{V_\tau}{\sum_{i=\tau}^{L} N_i} \right)}$$

- 3 approaches to obtain variance estimates:

1. Bound the covariances using **Cauchy-Schwarz**:
   analytic solution for the optimal number of samples required on each level, but the error bound is too conservative
2. Use the **de-biasing technique** from [Rhee, Glynn, 2015]:
   randomization of the final level L
3. Randomly shifted lattice rules from **Quasi-Monte Carlo**

Quasi-Monte Carlo (QMC)

- A **Quasi-Monte Carlo** method uses well-chosen sample points, as opposed to the random points with Monte Carlo
- A popular choice are **rank-1 lattice rules**

$$\boldsymbol{t}^{(n)} := \frac{n\boldsymbol{z} \bmod N}{N} = \left\{ \frac{n\boldsymbol{z}}{N} \right\}$$

where $\boldsymbol{z} \in \mathbb{Z}_N^s$ is a generating vector and $\{\cdot\}$ denotes mod 1

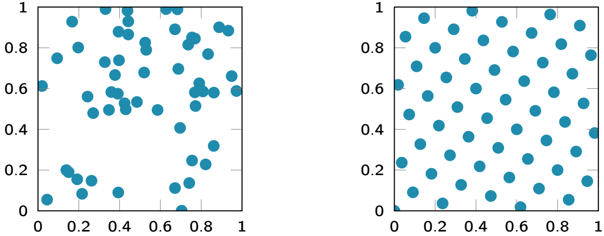
[1] more details to be found in standard works such as [Dick, Kuo, Sloan, 2013]

Quasi-Monte Carlo (QMC)

- A **Quasi-Monte Carlo** method uses well-chosen sample points, as opposed to the random points with Monte Carlo
- A popular choice are **rank-1 lattice rules**

$$\boldsymbol{t}^{(n)} := \frac{n\boldsymbol{z} \bmod N}{N} = \left\{ \frac{n\boldsymbol{z}}{N} \right\}$$

where $\boldsymbol{z} \in \mathbb{Z}_N^s$ is a generating vector and $\{\cdot\}$ denotes mod 1
- Can potentially obtain $\mathcal{O}(1/N)$ convergence, if integrand is *sufficiently smooth and decaying importance of dimensions*[1]



[1] more details to be found in standard works such as [Dick, Kuo, Sloan, 2013]

Random shifting

- Lattice points are chosen deterministically, hence, they are **correlated**
- Solution is **random shifting**:

$$\bar{\mathcal{Q}}^{\mathrm{QMC}}_{L,P,\mathrm{reuse}} := \frac{1}{P}\sum_{p=1}^{P}\sum_{\ell=0}^{L}\left(\frac{1}{\sum_{i=\ell}^{L} N_i}\right)\sum_{k=\ell}^{L}\sum_{n=1}^{N_k}\left(F_\ell(\boldsymbol{y}_{k,p}^{(n)}) - F_{\ell-1}(\boldsymbol{y}_{k,p}^{(n)})\right)$$

where $\boldsymbol{y}_{k,p}^{(n)} := \Phi^{-1}\left(\left\{\boldsymbol{t}_\ell^{(k)} + \boldsymbol{u}_k^{(p)}\right\}\right)$ and $\boldsymbol{u}_k^{(p)} \sim \boldsymbol{U}(0,1)$

Random shifting

- Lattice points are chosen deterministically, hence, they are **correlated**
- Solution is **random shifting**:

$$\bar{\mathcal{Q}}^{\text{QMC}}_{L,P,\text{reuse}} := \frac{1}{P}\sum_{p=1}^{P}\sum_{\ell=0}^{L}\left(\frac{1}{\sum_{i=\ell}^{L} N_i}\right)\sum_{k=\ell}^{L}\sum_{n=1}^{N_k}\left(F_\ell(\boldsymbol{y}_{k,p}^{(n)}) - F_{\ell-1}(\boldsymbol{y}_{k,p}^{(n)})\right)$$

where $\boldsymbol{y}_{k,p}^{(n)} := \Phi^{-1}\left(\left\{\boldsymbol{t}_\ell^{(k)} + \boldsymbol{u}_k^{(p)}\right\}\right)$ and $\boldsymbol{u}_k^{(p)} \sim \boldsymbol{U}(0,1)$

- Sample variance is used as an estimate for the variance

$$\mathbb{V}[\bar{\mathcal{Q}}^{\text{QMC}}_{L,P,\text{reuse}}] \approx \frac{1}{P(P-1)}\sum_{p=1}^{P}\left(\mathcal{Q}^{\text{QMC}}_{L,p,\text{reuse}} - \bar{\mathcal{Q}}^{\text{QMC}}_{L,P,\text{reuse}}\right)^2$$

Cost analysis

- Under the assumptions that

  (1) $|\mathrm{E}[Q_L - Q]| \leq c_\alpha h_L^\alpha$,

  (2) $V_\ell \leq c_\beta h_{\ell-1}^\beta$,

  (3) $C_\ell \leq c_\gamma h_\ell^{-\gamma}$, and

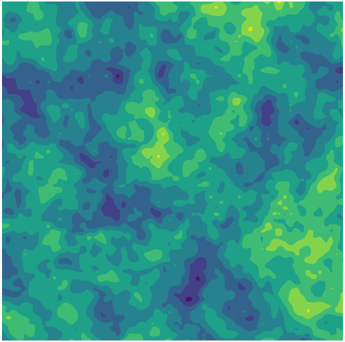
  (4) $\mathrm{V}[\bar{\mathcal{Y}}_\ell] \leq c_\lambda N_\ell^{-1/\lambda} V_\ell$.

  we can show that

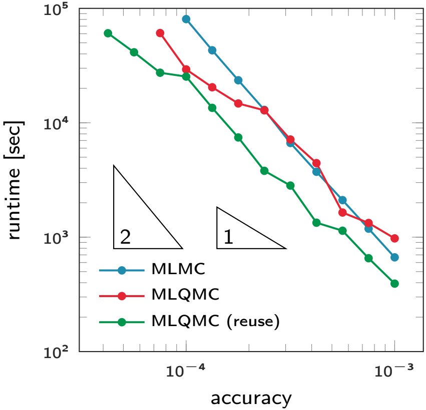
$$\mathrm{cost}(\bar{\mathcal{Q}}^{\mathrm{QMC}}_{L,P,\mathrm{reuse}}) = \left(1 - \left(s^{-(\beta+\gamma)}\right)^{\frac{\lambda}{\lambda+1}}\right) \mathrm{cost}(\bar{\mathcal{Q}}^{\mathrm{QMC}}_{L,P})$$

- This means that the sample reuse is more efficient when
  - The variance of the difference decays slowly (small β)
  - The lattice rule has good performance (small λ)

Numerical results

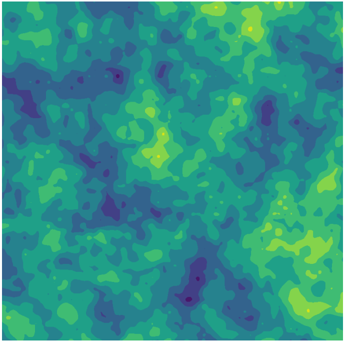


$\lambda = 0.1$, $\nu = 0.5$
3500 KL terms

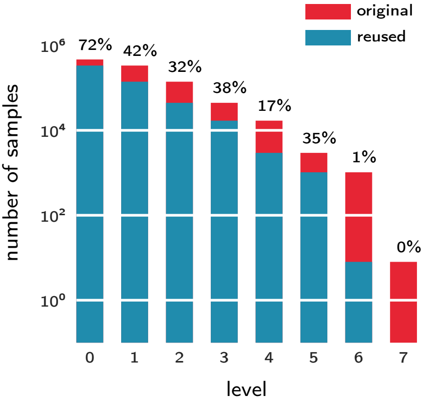


speedup vs MLMC:
- ~1.34 with MLQMC
- ~2.59 with MLQMC (reuse)

Numerical results

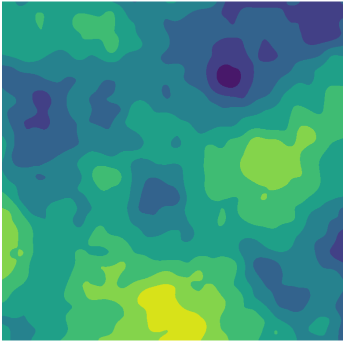


$\lambda = 0.1$, $\nu = 0.5$
3500 KL terms

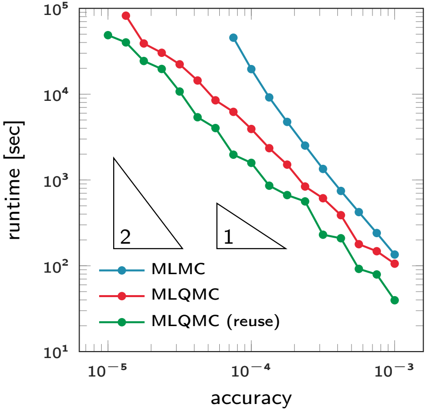


~$3 \cdot 10^6$ samples on a 4×4 grid
~$2 \cdot 10^2$ samples on a 512×512 grid

Numerical results



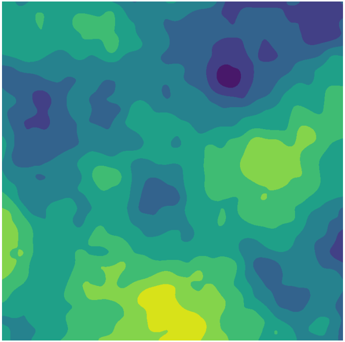
$\lambda = 0.3$, $\nu = 1$
1000 KL terms



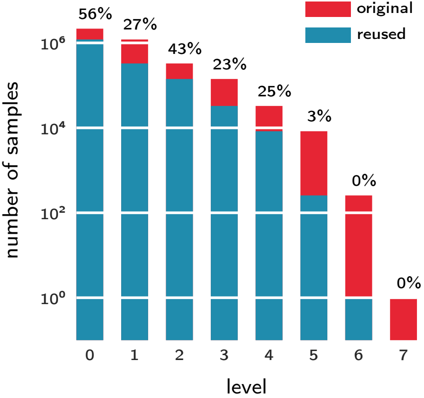
speedup vs MLMC:

~3.17 with MLQMC
~7.82 with MLQMC (reuse)

Numerical results

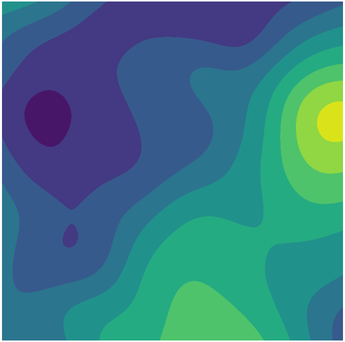


$\lambda = 0.3$, $\nu = 1$
1000 KL terms

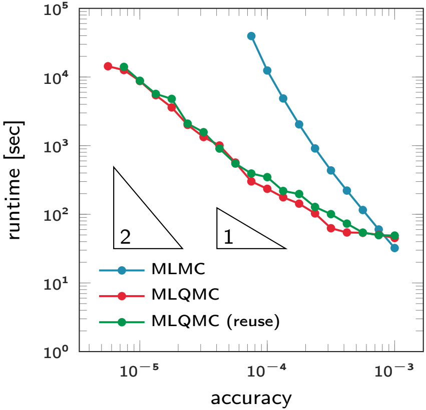


~$2 \cdot 10^7$ samples on a 4×4 grid
~$2 \cdot 10^1$ samples on a 512×512 grid

Numerical results



$\lambda = 0.5$, $\nu = 2$
100 KL terms

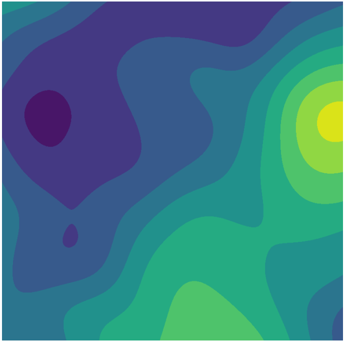


speedup vs MLMC:

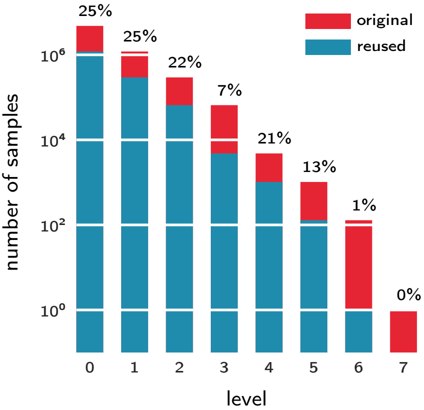
~41.93 with MLQMC

~36.94 with MLQMC (reuse)

Numerical results



$\lambda = 0.5$, $\nu = 2$
100 KL terms



~$7 \cdot 10^7$ samples on a 4×4 grid
~$2 \cdot 10^1$ samples on a 512×512 grid

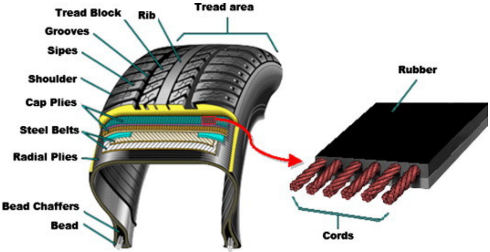
PART 3

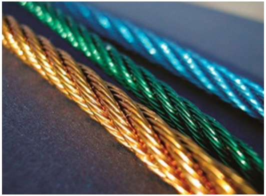
Application: Wire drawing and Bekaert

Bekaert

BEKAERT
better together

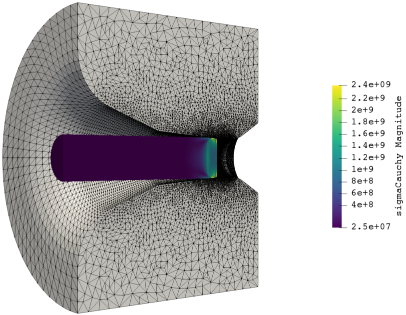
- Belgian company, est. 1880
- Steel wire transformation and coatings
- 30.000 people in 120 countries
- Physical Modelling Team

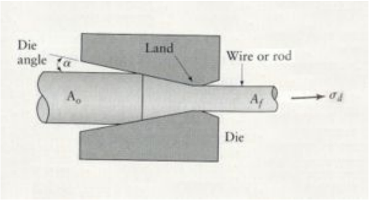




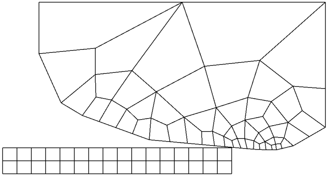
Wire drawing test case

- 2d axisymmetric geometry with die and wire
- **22 uncertainties**: geometrical, physical, process-related. . .
- Quantity of interest: drawing force, stress distribution after several drawing passes

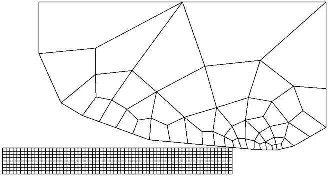




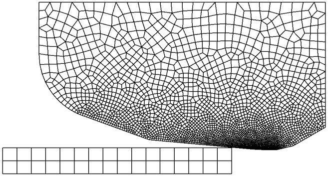
Selection of coarse approximations



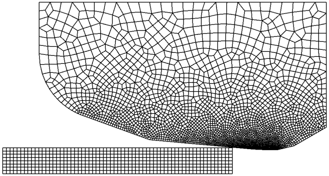
(0,0)



(0,2)

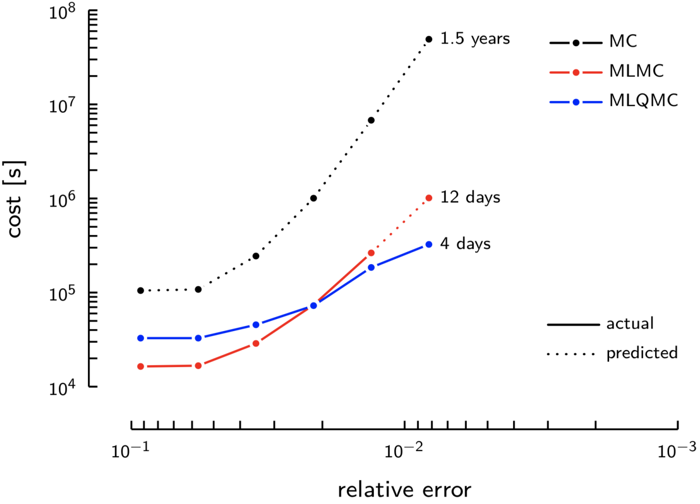


(3,0)



(3,2)

Results



10^8
10^7
10^6
10^5
10^4
cost [s]
1.5 years
12 days
4 days
MC
MLMC
MLQMC
actual
predicted
10^{-1}
10^{-2}
10^{-3}
relative error

Conclusions

- **MLMC** is an efficient **variance reduction** technique
  - Estimate differences between subsequent approximations and exploit telescoping sum to obtain cost reduction
  - Most samples are taken on the coarse grids, and only few samples are required on the finest grid
  - All benefits of Monte Carlo methods remain

- Discussed the **MG-MLQMC** extension
  - FMG solver yields free samples on coarse grids
  - Can be reused if care is taken not to introduce additional statistical error ($\rightarrow$ random shifting from QMC)

- Application to real-life engineering problem
  - Can couple with existing code without much effort
  - Large gain by using the MLQMC method

Conclusions

- **MLMC** is an efficient **variance reduction** technique
  - Estimate differences between subsequent approximations and exploit telescoping sum to obtain cost reduction
  - Most samples are taken on the coarse grids, and only few samples are required on the finest grid
  - All benefits of Monte Carlo methods remain

- Discussed the **MG-MLQMC** extension
  - FMG solver yields free samples on coarse grids
  - Can be reused if care is taken not to introduce additional statistical error ($\rightarrow$ random shifting from QMC)

- Application to real-life engineering problem
  - Can couple with existing code without much effort
  - Large gain by using the MLQMC method

Thank you for your attention