Multi-level Monte Carlo methods in Uncertainty Quantification

Fabio Nobile

CSQI - Institute of Mathematics, EPFL, Switzerland


3rd GAMM AGUQ workshop on Uncertainty Quantification Dortmund, March 12-14, 2018

EU-FP7 project: Uncertainty Management for Robust Industrial Design in Aeronautics (UMRIDA)

Center for Advanced Modeling Science
Outline

1 Motivating example

2 Multilevel Monte Carlo for expectations

3 MLMC for moments and distributions

4 Risk averse optimization with MLMC

5 Conclusions
Motivating example

**UQ in aerodynamic design**

Compute aerodynamic coeffs. (lift, drag, $C_p$) and optimize airfoil shape in presence of operational uncertainties (Mach number, angle of attack, ...) and geometrical uncertainties (manufacturing tolerances, icing, fatigue, ...)

RAE2882
Operational uncertainties

Atmospheric fluctuations with respect to location, time ($T, p, \rho, u$) over long flights

**Temperature [K] - ground - 1/JAN/2015**

**Temperature [K] - ground - 1/JUL/2015**

**U wind [m/s] - ground - 1/JUL/2015**

**V wind [m/s] - ground - 1/JUL/2015**

**Probabilistic framework**: Mach, Reynolds, Angle of Attack, etc. treated as random variables
Geometrical uncertainties

**Production:** manufacturing, assembly

**Temporary factors:** deflection, icing

**Permanent/degrading factors:** impacts, erosion, fouling

**Probabilistic Framework:** Leading edge radius, thickness, curvature, etc. treated as random variables
Forward Uncertainty propagation

- **Random input parameters:** $y$ (with given distribution)
- **(Complex) Model:** $\mathcal{L}_y u = \mathcal{F}$ (e.g. Euler, Navier-Stokes,...)
  hence $u = u(y)$ is a random solution
- **Quantity of interest:** $Q = Q(u)$ (random output, e.g. lift, drag, etc.)

**Goal:** compute $\mu(Q) = \mathbb{E}[Q]$ or other statistical quantities

In practice, $u$ is not accessible. **Computational model**

$$\mathcal{L}_{h,y} u_h = \mathcal{F}_h \quad \implies \quad \text{computational output} \quad Q_h = Q(u_h)$$
Forward Uncertainty propagation

- Random input parameters: $y$ (with given distribution)
- (Complex) Model: $\mathcal{L}_y u = \mathcal{F}$ (e.g. Euler, Navier-Stokes,...)
  hence $u = u(y)$ is a random solution
- Quantity of interest: $Q = Q(u)$ (random output, e.g. lift, drag, etc.)

Goal: compute $\mu(Q) = \mathbb{E}[Q]$ or other statistical quantities

In practice, $u$ is not accessible. Computational model

$$\mathcal{L}_{h,y} u_h = \mathcal{F}_h \quad \implies \quad \text{computational output} \quad Q_h = Q(u_h)$$
Forward Uncertainty propagation

- Random input parameters: $y$ (with given distribution)
- (Complex) Model: $\mathcal{L}_y u = \mathcal{F}$ (e.g. Euler, Navier-Stokes,...)
  hence $u = u(y)$ is a random solution
- Quantity of interest: $Q = Q(u)$ (random output, e.g. lift, drag, etc.)

Goal: compute $\mu(Q) = \mathbb{E}[Q]$ or other statistical quantities

In practice, $u$ is not accessible. Computational model

$$\mathcal{L}_{h,y} u_h = \mathcal{F}_h \quad \implies \quad \text{computational output} \quad Q_h = Q(u_h)$$
Monte Carlo method

- Generate $M$ iid copies $y^{(1)}, \ldots, y^{(M)} \sim y$
- Compute the corresponding outputs $Q^{(i)}_h$, $i = 1, \ldots, M$
- Approximate expectation by sample average

$$
\mu^\text{MC}_h = \frac{1}{M} \sum_{i=1}^{M} Q^{(i)}_h \\
\text{(biased estimator } \mathbb{E}[\mu^\text{MC}_h] = \mathbb{E}[Q_h] \neq \mathbb{E}[Q])
$$

Mean squared error

$$
\text{MSE}(\mu^\text{MC}_h) := \mathbb{E}[(\mu(Q) - \mu^\text{MC}_h)^2] = \left(\mathbb{E}[Q - Q_h]\right)^2 + \frac{\text{Var}[Q_h]}{M}
$$

Complexity analysis (error versus cost)

Assume:
- $|\mathbb{E}[Q - Q_h]| = \mathcal{O}(h^{\alpha})$, $\text{Var}[Q_h] = \mathcal{O}(1)$,
- cost to compute each $Q^{(i)}_h$: $C_h = \mathcal{O}(h^{-\gamma})$

Then

$$
\text{MSE} = \mathcal{O}(\text{tol}^2) \implies h = \mathcal{O}(\text{tol}^{\frac{1}{\alpha}}), \quad M = \mathcal{O}(\text{tol}^{-2})
$$

Total work: $\text{Work}(\mu^\text{MC}_h) = C_h M \lesssim \text{tol}^{-\frac{\gamma}{\alpha}} \text{tol}^{-2}$
Monte Carlo method

- Generate $M$ iid copies $y^{(1)}, \ldots, y^{(M)} \sim y$
- Compute the corresponding outputs $Q_h^{(i)}$, $i = 1, \ldots, M$
- Approximate expectation by sample average

$$
\mu_h^{MC} = \frac{1}{M} \sum_{i=1}^{M} Q_h^{(i)} \quad \text{(biased estimator } \mathbb{E}[\mu_h^{MC}] = \mathbb{E}[Q_h] \neq \mathbb{E}[Q])
$$

Mean squared error

$$
\text{MSE}(\mu_h^{MC}) := \mathbb{E}[(\mu(Q) - \mu_h^{MC})^2] = \left(\mathbb{E}[Q - Q_h]\right)^2 + \frac{\text{Var}[Q_h]}{M}
$$

Complexity analysis (error versus cost)

Assume:

- $|\mathbb{E}[Q - Q_h]| = \mathcal{O}(h^\alpha)$, $\text{Var}[Q_h] = \mathcal{O}(1)$,
- cost to compute each $Q_h^{(i)}$: $C_h = \mathcal{O}(h^{-\gamma})$

Then

$$
\text{MSE} = \mathcal{O}(\text{tol}^2) \quad \implies \quad h = \mathcal{O}(\text{tol}^{\frac{1}{\alpha}}), \quad M = \mathcal{O}(\text{tol}^{-2})
$$

Total work:

$$
\text{Work}(\mu_h^{MC}) = C_h M \lesssim \text{tol}^{-\frac{\gamma}{\alpha}} \text{tol}^{-2}
$$
Monte Carlo method

- Generate $M$ iid copies $y^{(1)}, \ldots, y^{(M)} \sim y$
- Compute the corresponding outputs $Q_h^{(i)}$, $i = 1, \ldots, M$
- Approximate expectation by sample average

$$\mu_h^{MC} = \frac{1}{M} \sum_{i=1}^{M} Q_h^{(i)} \quad \text{(biased estimator $\mathbb{E}[\mu_h^{MC}] = \mathbb{E}[Q_h] \neq \mathbb{E}[Q]$)}$$

Mean squared error

$$\text{MSE}(\mu_h^{MC}) := \mathbb{E}[(\mu(Q) - \mu_h^{MC})^2] = \left(\mathbb{E}[Q - Q_h]\right)^2 + \frac{\text{Var}[Q_h]}{M}$$

Complexity analysis (error versus cost)

Assume:

- $|\mathbb{E}[Q - Q_h]| = \mathcal{O}(h^\alpha)$, $\text{Var}[Q_h] = \mathcal{O}(1)$,
- cost to compute each $Q_h^{(i)}$: $C_h = \mathcal{O}(h^{-\gamma})$

Then

$$\text{MSE} = \mathcal{O}(tol^2) \quad \implies \quad h = \mathcal{O}(tol^{\frac{1}{\alpha}}), \quad M = \mathcal{O}(tol^{-2})$$

Total work:

$$\text{Work}(\mu_h^{MC}) = C_h M \lesssim tol^{-\frac{\gamma}{\alpha}} tol^{-2}$$
Can we improve on Monte Carlo? Control variate

Let $Z$ be random variable correlated with $Q_h$, and with known mean.

**Idea:** Apply MC on $Q_{h,Z} = Q_h - \alpha(Z - \mathbb{E}[Z])$ (notice that $\mathbb{E}[Q_{h,Z}] = \mathbb{E}[Q_h]$)

$$\mu_{h,\text{CV}} = \frac{1}{M} \sum_{i=1}^{M} (Q_h - \alpha Z + \alpha \mathbb{E}[Z])$$

$$\text{Var}[Q_{h,Z}] = \text{Var}[Q_h - \alpha Z] = \text{Var}[Q_h] + \alpha^2 \text{Var}[Z] - 2 \alpha \text{Cov}(Q_h, Z)$$

For optimal $\alpha$: $\text{Var}[Q_{h,Z}] = \text{Var}[Q_h] \left(1 - \frac{\text{Cov}(Q_h, Z)}{\text{Var}[Z]}\right) \leq \text{Var}[Q_h]$ (always gives variance reduction)

**Two ideas for choosing $Z$**

- Use a surrogate model $Z = Q_{surr}$ with numerically optimized $\alpha$
  $\Rightarrow$ *multi-fidelity Monte Carlo* [Peherstorfer, Willcox, Gunzburger, 2016]

- Use coarser discretization e.g. $Z = Q_{2h}$ (usually with $\alpha = 1$)
  $\Rightarrow$ *two level Monte Carlo* [Heinrich 1998, Giles 2008, ...]
Can we improve on Monte Carlo? Control variate

Let $Z$ be random variable correlated with $Q_h$, and with known mean.

**Idea:** Apply MC on $Q_{h,Z} = Q_h - \alpha(Z - \mathbb{E}[Z])$ (notice that $\mathbb{E}[Q_{h,Z}] = \mathbb{E}[Q_h]$)

$$\mu_{h}^{CV} = \frac{1}{M} \sum_{i=1}^{M} (Q_h^{(i)} - \alpha Z^{(i)}) + \alpha \mathbb{E}[Z]$$

$$\text{Var}[Q_{h,Z}] = \text{Var}[Q_h - \alpha Z] = \text{Var}[Q_h] + \alpha^2 \text{Var}[Z] - 2\alpha \text{Cov}(Q_h, Z)$$

For optimal $\alpha$: $\text{Var}[Q_{h,Z}] = \text{Var}[Q_h] \left(1 - \frac{\text{Cov}(Q_h, Z)}{\text{Var}[Z]}\right) \leq \text{Var}[Q_h]$ (always gives variance reduction)

Two ideas for choosing $Z$

- Use a surrogate model $Z = Q_{surr}$ with numerically optimized $\alpha$
  $\implies$ multi-fidelity Monte Carlo [Peherstorfer, Willcox, Gunzburger, 2016]
- Use coarser discretization e.g. $Z = Q_{2h}$ (usually with $\alpha = 1$)
  $\implies$ two level Monte Carlo [Heinrich 1998, Giles 2008, ...]
Can we improve on Monte Carlo? Control variate

Let $Z$ be random variable correlated with $Q_h$, and with known mean.

Idea: Apply MC on $Q_h, Z = Q_h - \alpha (Z - \mathbb{E}[Z])$ (notice that $\mathbb{E}[Q_h, Z] = \mathbb{E}[Q_h]$)

$$\mu_h^{CV} = \frac{1}{M} \sum_{i=1}^{M} (Q_h(i) - \alpha Z(i)) + \alpha \mathbb{E}[Z]$$

$$\text{Var}[Q_h, Z] = \text{Var}[Q_h - \alpha Z] = \text{Var}[Q_h] + \alpha^2 \text{Var}[Z] - 2\alpha \text{Cov}(Q_h, Z)$$

For optimal $\alpha$: $\text{Var}[Q_h, Z] = \text{Var}[Q_h] \left(1 - \frac{\text{Cov}(Q_h, Z)}{\text{Var}[Z]}\right) \leq \text{Var}[Q_h]$ (always gives variance reduction)

Two ideas for choosing $Z$

- Use a surrogate model $Z = Q^{\text{surr}}$ with numerically optimized $\alpha$
  $\Rightarrow$ multi-fidelity Monte Carlo [Peherstorfer, Willcox, Gunzburger, 2016]

- Use coarser discretization e.g. $Z = Q_{2h}$ (usually with $\alpha = 1$)
  $\Rightarrow$ two level Monte Carlo [Heinrich 1998, Giles 2008, ...]
Can we improve on Monte Carlo? Control variate

**Problem:** $\mathbb{E}[Z]$ not known, in general!

$\implies$ compute it with independent MC with larger sample size (cheaper problem).

From two-level to multilevel:

$$
\mu^C_h = \frac{1}{M} \sum_{i=1}^{M} (Q_h^{(i)} - Q_{2h}^{(i)}) + \mathbb{E}[Q_{2h}]
$$

$$
\approx \frac{1}{M} \sum_{i=1}^{M} (Q_h^{(i)} - Q_{2h}^{(i)}) + \frac{1}{M_2} \sum_{i=1}^{M_2} Q_{2h}^{(i,2)}, \quad M_2 > M
$$

$$
\approx \frac{1}{M} \sum_{i=1}^{M} (Q_h^{(i)} - Q_{2h}^{(i)}) + \frac{1}{M_2} \sum_{i=1}^{M_2} (Q_{2h}^{(i,2)} - Q_{4h}^{(i,2)}) + \ldots + \frac{1}{M_n} \sum_{i=1}^{M_n} Q_{2^{n}h}^{(i,n)}
$$
Can we improve on Monte Carlo? Control variate

**Problem:** $\mathbb{E}[Z]$ not known, in general!

$\leadsto$ compute it with independent MC with larger sample size (cheaper problem).

From two-level to multilevel:

$$
\mu_h^{CV} = \frac{1}{M} \sum_{i=1}^{M} (Q_h^{(i)} - Q_{2h}^{(i)}) + \mathbb{E}[Q_{2h}]
$$

$$
\approx \frac{1}{M} \sum_{i=1}^{M} (Q_h^{(i)} - Q_{2h}^{(i)}) + \frac{1}{M_2} \sum_{i=1}^{M_2} Q_{2h}^{(i,2)}, \quad M_2 > M
$$

$$
\approx \frac{1}{M} \sum_{i=1}^{M} (Q_h^{(i)} - Q_{2h}^{(i)}) + \frac{1}{M_2} \sum_{i=1}^{M_2} (Q_{2h}^{(i,2)} - Q_{4h}^{(i,2)}) + \ldots + \frac{1}{M_n} \sum_{i=1}^{M_n} Q_{2^n h}^{(i,n)}
$$
Motivating example

Can we improve on Monte Carlo? Control variate

**Problem:** $\mathbb{E}[Z]$ not known, in general!

$\leadsto$ compute it with independent MC with larger sample size (cheaper problem).

**From two-level to multilevel:**

\[
\mu_{h}^{CV} = \frac{1}{M} \sum_{i=1}^{M} (Q_{h}^{(i)} - Q_{2h}^{(i)}) + \mathbb{E}[Q_{2h}]
\]

\[
\simeq \frac{1}{M} \sum_{i=1}^{M} (Q_{h}^{(i)} - Q_{2h}^{(i)}) + \frac{1}{M_2} \sum_{i=1}^{M_2} Q_{2h}^{(i,2)}, \quad M_2 > M
\]

\[
\simeq \frac{1}{M} \sum_{i=1}^{M} (Q_{h}^{(i)} - Q_{2h}^{(i)}) + \frac{1}{M_2} \sum_{i=1}^{M_2} (Q_{2h}^{(i,2)} - Q_{4h}^{(i,2)}) + \ldots + \frac{1}{M_n} \sum_{i=1}^{M_n} Q_{2^n h}^{(i,n)}
\]
Can we improve on Monte Carlo? Control variate

**Problem:** $\mathbb{E}[Z]$ not known, in general!

$\leadsto$ compute it with independent MC with larger sample size (cheaper problem).

From two-level to multilevel:

$$
\mu_{h}^{CV} = \frac{1}{M} \sum_{i=1}^{M} (Q_{h}^{(i)} - Q_{2h}^{(i)}) + \mathbb{E}[Q_{2h}]
$$

$$
\leadsto \frac{1}{M} \sum_{i=1}^{M} (Q_{h}^{(i)} - Q_{2h}^{(i)}) + \frac{1}{M_2} \sum_{i=1}^{M_2} Q_{2h}^{(i,2)}, \quad M_2 > M
$$

$$
\leadsto \frac{1}{M} \sum_{i=1}^{M} (Q_{h}^{(i)} - Q_{2h}^{(i)}) + \frac{1}{M_2} \sum_{i=1}^{M_2} (Q_{2h}^{(i,2)} - Q_{4h}^{(i,2)}) + \ldots + \frac{1}{M_n} \sum_{i=1}^{M_n} Q_{2^n h}^{(i,n)}
$$
Outline

1 Motivating example

2 Multilevel Monte Carlo for expectations

3 MLMC for moments and distributions

4 Risk averse optimization with MLMC

5 Conclusions
Multilevel Monte Carlo

- Sequence of refined discretizations
  \[ h_0 > h_1 > \ldots > h_L \]
- Sequence of sample sizes
  \[ M_0 > M_1 > \ldots > M_L \]

Denoting \( Q_\ell = Q_{h_\ell} \), the MLMC estimator is

\[
\mu_{L}^{\text{MLMC}} = \sum_{\ell=0}^{L} \frac{1}{M_\ell} \sum_{i=1}^{M_\ell} \left( Q_\ell^{(i,\ell)} - Q_{\ell-1}^{(i,\ell)} \right), \quad Q_{-1} = 0
\]

Mean squared error

\[
\text{MSE}(\mu_{L}^{\text{MLMC}}) = \left( \mathbb{E}[Q] - Q_L \right)^2 + \sum_{\ell=0}^{L} \frac{\text{Var}[Q_\ell - Q_{\ell-1}]}{M_\ell}
\]

\( \text{discret. error level } L \)\( \text{statistical error} \)
Multilevel Monte Carlo

Sequence of refined discretizations

\[ h_0 > h_1 > \ldots > h_L \]

Sequence of sample sizes

\[ M_0 > M_1 > \ldots > M_L \]

Denoting \( Q_\ell = Q_{h_\ell} \), the MLMC estimator is

\[
\mu_{L}^{\text{MLMC}} = \frac{1}{M_0} \sum_{i=1}^{M_0} Q_0^{(i,0)} + \frac{1}{M_1} \sum_{i=1}^{M_1} \left( Q_1^{(i,1)} - Q_0^{(i,1)} \right) + \ldots + \frac{1}{M_L} \sum_{i=1}^{M_L} \left( Q_L^{(i,L)} - Q_{L-1}^{(i,L)} \right)
\]

Mean squared error

\[
\text{MSE}(\mu_{L}^{\text{MLMC}}) = \left( \mathbb{E}[Q - Q_L] \right)^2 + \sum_{\ell=0}^{L} \frac{\text{Var}[Q_\ell - Q_{\ell-1}]}{M_\ell}
\]

\( \text{discret. error level } L \)

\( \text{statistical error} \)
Multilevel Monte Carlo

- Sequence of refined discretizations
  \[ h_0 > h_1 > \ldots > h_L \]
- Sequence of sample sizes
  \[ M_0 > M_1 > \ldots > M_L \]

Denoting \( Q_\ell = Q_{h_\ell} \), the MLMC estimator is

\[
\mu_{L}^{\text{MLMC}} = \sum_{\ell=0}^{L} \frac{1}{M_\ell} \sum_{i=1}^{M_\ell} (Q^{(i,\ell)} - Q^{(i,\ell-1)}), \quad Q_{-1} = 0
\]

Mean squared error

\[
\text{MSE}(\mu_{L}^{\text{MLMC}}) = \left( \mathbb{E}[Q - Q_L] \right)^2 + \sum_{\ell=0}^{L} \frac{\text{Var}[Q_\ell - Q_{\ell-1}]}{M_\ell}
\]

discret. error level \( L \)

statistical error
Multilevel Monte Carlo

- Sequence of refined discretizations
  \[ h_0 > h_1 > \ldots > h_L \]
- Sequence of sample sizes
  \[ M_0 > M_1 > \ldots > M_L \]

Denoting \( Q_\ell = Q_{h_\ell} \), the MLMC estimator is

\[
\mu_{L}^{\text{MLMC}} = \sum_{\ell=0}^{L} \frac{1}{M_\ell} \sum_{i=1}^{M_\ell} (Q_{\ell}^{(i,\ell)} - Q_{\ell-1}^{(i,\ell)}), \quad Q_{-1} = 0
\]

Mean squared error

\[
\text{MSE}(\mu_{L}^{\text{MLMC}}) = \left( \mathbb{E}[Q - Q_L] \right)^2 + \sum_{\ell=0}^{L} \frac{\text{Var}[Q_{\ell} - Q_{\ell-1}]}{M_\ell}
\]

- Discret. error level \( L \)
- Statistical error
Multilevel Monte Carlo

- $V_\ell = \text{Var}[Q_\ell - Q_{\ell-1}]$ (variance of differences)
- $C_\ell = \text{cost of computing each } \Delta Q_\ell^{(i,\ell)} = Q_\ell^{(i,\ell)} - Q_{\ell-1}^{(i,\ell)}$

Optimal sample sizes $M_\ell$: [Giles 2008] minimize $W = \sum_{\ell=0}^{L} C_\ell M_\ell$ s.t. $\text{MSE} \sim tol^2$

$$M_\ell = \left\lceil tol^{-2} \sqrt{\frac{V_\ell}{C_\ell}} \left( \sum_{k=0}^{L} \sqrt{C_k V_k} \right) \right\rceil$$

Complexity analysis for $h_\ell = h_0 s^{-\ell}$: [Giles 2008, Cliffe-Giles-Scheichl-Teckentrup 2011]

Assume

- $|\mathbb{E}[Q - Q_\ell]| = O(h_\ell^\alpha)$,
- $V_\ell = \text{Var}[Q_\ell - Q_{\ell-1}] = O(h_\ell^\beta)$, \hspace{1em} ($\beta = 2\alpha$ for smooth problems/noise)
- $C_\ell = O(h_\ell^{-\gamma})$, \hspace{1em} $2\alpha \geq \min\{\beta, \gamma\}$

Then, choosing $L = O(tol^{\frac{1}{\alpha}})$ and $M_\ell$ as above gives $\text{MSE}(\mu_L^{\text{MLMC}}) \leq tol^2$ and

$$\text{Work}(\mu_L^{\text{MLMC}}) = \sum_{\ell=0}^{L} C_\ell M_\ell \lesssim \begin{cases} 
tol^{-2}, & \beta > \gamma \\
tol^{-2}(\log tol)^2, & \beta = \gamma \\
tol^{-2} - \frac{\gamma - \beta}{\alpha}, & \beta < \gamma 
\end{cases}$$
Multilevel Monte Carlo

- $V_\ell = \text{Var}[Q_\ell - Q_{\ell-1}]$ (variance of differences)
- $C_\ell = \text{cost of computing each } \Delta Q_\ell^{(i,\ell)} = Q_\ell^{(i,\ell)} - Q_{\ell-1}^{(i,\ell)}$

Optimal sample sizes $M_\ell$: [Giles 2008] minimize $W = \sum_{\ell=0}^{L} C_\ell M_\ell$ s.t. $\text{MSE} \simeq tol^2$

$$M_\ell = \left\lfloor tol^{-2} \sqrt{\frac{V_\ell}{C_\ell}} \left( \sum_{k=0}^{L} \sqrt{C_k V_k} \right) \right\rfloor$$

Complexity analysis for $h_\ell = h_0 s^{-\ell}$: [Giles 2008, Cliffe-Giles-Scheichl-Teckentrup 2011]

Assume

- $|E[Q - Q_\ell]| = O(h_\ell^\alpha)$,
- $V_\ell = \text{Var}[Q_\ell - Q_{\ell-1}] = O(h_\ell^\beta)$, \hspace{1cm} ($\beta = 2\alpha$ for smooth problems/noise)
- $C_\ell = O(h_\ell^{-\gamma})$, \hspace{1cm} $2\alpha \geq \min\{\beta, \gamma\}$

Then, choosing $L = O(tol^{\frac{1}{\alpha}})$ and $M_\ell$ as above gives $\text{MSE}(\mu_L^{\text{MLMC}}) \leq tol^2$ and

$$\text{Work}(\mu_L^{\text{MLMC}}) = \sum_{\ell=0}^{L} C_\ell M_\ell \lesssim \begin{cases} tol^{-2}, & \beta > \gamma \\ tol^{-2}(\log tol)^2, & \beta = \gamma \\ tol^{-2-\frac{\gamma-\beta}{\alpha}}, & \beta < \gamma \end{cases}$$
Multilevel Monte Carlo

- $V_\ell = \text{Var}[Q_\ell - Q_{\ell-1}]$ (variance of differences)
- $C_\ell = \text{cost of computing each } \Delta Q^{(i,\ell)}_\ell = Q^{(i,\ell)}_\ell - Q^{(i,\ell)}_{\ell-1}$

Optimal sample sizes $M_\ell$: [Giles 2008] minimize $W = \sum_{\ell=0}^{L} C_\ell M_\ell$ s.t. $\text{MSE} \simeq tol^2$

$$M_\ell = \left[ tol^{-2} \sqrt{\frac{V_\ell}{C_\ell}} \left( \sum_{k=0}^{L} \sqrt{C_k V_k} \right) \right]$$

Complexity analysis for $h_\ell = h_0 s^{-\ell}$: [Giles 2008, Cliffe-Giles-Scheichl-Teckentrup 2011]

Assume

- $|E[Q - Q_\ell]| = O(h_\ell^\alpha)$,
- $V_\ell = \text{Var}[Q_\ell - Q_{\ell-1}] = O(h_\ell^\beta)$, \hspace{1cm} ($\beta = 2\alpha$ for smooth problems/noise)
- $C_\ell = O(h_\ell^{-\gamma})$, \hspace{1cm} $2\alpha \geq \min\{\beta, \gamma\}$

Then, choosing $L = O(tol^{\frac{1}{\alpha}})$ and $M_\ell$ as above gives $\text{MSE}(\mu_L^{\text{MLMC}}) \leq tol^2$ and

$$\text{Work}(\mu_L^{\text{MLMC}}) = \sum_{\ell=0}^{L} C_\ell M_\ell \lesssim \begin{cases} tol^{-2}, & \beta > \gamma \\ tol^{-2}(\log tol)^2, & \beta = \gamma \\ tol^{-2} - \frac{\gamma - \beta}{\alpha}, & \beta < \gamma \end{cases}$$
Multilevel Monte Carlo – practical aspects

**Remark**: MC complexity always improved for optimal choice of $M_{\ell}$. For $\beta = 2\alpha$ we get either $O(tol^{-2})$ (up to log terms) or $O(tol^{-\frac{\gamma}{\alpha}})$.

To achieve improved complexity, one needs to
- estimate error decay $|\mathbb{E}[Q - Q_{\ell}]|$: $\sim$ needed to determine optimal $L$
- estimate variance decay $V_{\ell}$: $\sim$ needed to determine optimal $\{M_{\ell}\}_{\ell=0}^L$

$|\mathbb{E}[Q - Q_{\ell}]|$ can be estimated as $|\mu_{\ell}^{MC} - \mu_{\ell-1}^{MC}|$ based on a pilot run.

$V_{\ell}$ can be estimated by sample variance estimator based on pilot runs.

**Problem**: on the finest levels we should run only very few simulations.

Cost for estimation of $V_L$ might dominate the overall cost of the MLMC algorithm.

**Idea**: use adaptive algorithms: extrapolate information from previous levels and correct it when samples become available.
Multilevel Monte Carlo – practical aspects

**Remark:** MC complexity always improved for optimal choice of $M_\ell$. For $\beta = 2\alpha$ we get either $O(tol^{-2})$ (up to log terms) or $O(tol^{-\frac{\gamma}{\alpha}})$.

To achieve improved complexity, one needs to
- estimate error decay $|E[Q - Q_\ell]|$: $\rightsquigarrow$ needed to determine optimal $L$
- estimate variance decay $V_\ell$: $\rightsquigarrow$ needed to determine optimal $\{M_\ell\}_{\ell=0}^L$

$|E[Q - Q_\ell]|$ can be estimated as $|\mu_{MC_\ell} - \mu_{MC_{\ell-1}}|$ based on a pilot run. $V_\ell$ can be estimated by sample variance estimator based on pilot runs.

**Problem:** on the finest levels we should run only very few simulations. Cost for estimation of $V_L$ might dominate the overall cost of the MLMC algorithm.

**Idea:** use adaptive algorithms: extrapolate information from previous levels and correct it when samples become available.
Remark: MC complexity always improved for optimal choice of $M_\ell$. For $\beta = 2\alpha$ we get either $O(tol^{-2})$ (up to log terms) or $O(tol^{-\frac{\gamma}{\alpha}})$.

To achieve improved complexity, one needs to

- estimate error decay $|E[Q - Q_\ell]|$: $\xrightarrow{}$ needed to determine optimal $L$
- estimate variance decay $V_\ell$: $\xrightarrow{}$ needed to determine optimal $\{M_\ell\}_{\ell=0}^L$

$|E[Q - Q_\ell]|$ can be estimated as $|\mu_\ell^{MC} - \mu_{\ell-1}^{MC}|$ based on a pilot run

$V_\ell$ can be estimated by sample variance estimator based on pilot runs

Problem: on the finest levels we should run only very few simulations. Cost for estimation of $V_L$ might dominate the overall cost of the MLMC algorithm.

Idea: use adaptive algorithms: extrapolate information from previous levels and correct it when samples become available.
Remark: MC complexity always improved for optimal choice of $M_\ell$. For $\beta = 2\alpha$ we get either $O(tol^{-2})$ (up to log terms) or $O(tol^{-\frac{\gamma}{\alpha}})$.

To achieve improved complexity, one needs to

- estimate error decay $|E[Q - Q_\ell]|$: $\rightsquigarrow$ needed to determine optimal $L$
- estimate variance decay $V_\ell$: $\rightsquigarrow$ needed to determine optimal $\{M_\ell\}_{\ell=0}^L$

$|E[Q - Q_\ell]|$ can be estimated as $|\mu_{M_\ell}^{MC} - \mu_{M_{\ell-1}}^{MC}|$ based on a pilot run.

$V_\ell$ can be estimated by sample variance estimator based on pilot runs.

Problem: on the finest levels we should run only very few simulations. Cost for estimation of $V_L$ might dominate the overall cost of the MLMC algorithm.

Idea: use adaptive algorithms: extrapolate information from previous levels and correct it when samples become available.
Multilevel Monte Carlo – practical aspects

**Remark**: MC complexity always improved for optimal choice of $M_\ell$. For $\beta = 2\alpha$ we get either $O(tol^{-2})$ (up to log terms) or $O(tol^{-\gamma/\alpha})$.

To achieve improved complexity, one needs to

- estimate error decay $|E[Q - Q_\ell]|$: $\leadsto$ needed to determine optimal $L$
- estimate variance decay $V_\ell$: $\leadsto$ needed to determine optimal $\{M_\ell\}_{\ell=0}^L$

$|E[Q - Q_\ell]|$ can be estimated as $|\mu_{\ell}^{MC} - \mu_{\ell-1}^{MC}|$ based on a pilot run $V_\ell$ can be estimated by sample variance estimator based on pilot runs

**Problem**: on the finest levels we should run only very few simulations. Cost for estimation of $V_L$ might dominate the overall cost of the MLMC algorithm.

**Idea**: use adaptive algorithms: extrapolate information from previous levels and correct it when samples become available.
Continuation Multilevel Monte Carlo


**Idea**: Solve the problem with decreasing tolerances $tol^{(0)} > tol^{(1)} > \ldots \geq tol$. Use collected samples on all levels to improve the estimate of $V_\ell$ and $|\mathbb{E}[Q - Q_\ell]|$.

Estimator $\hat{V}_\ell$ of $V_\ell = \text{Var}[\Delta Q_\ell]$ at iteration $j$: MAP Bayesian estimator

- we make the ansatz $\Delta Q_\ell \sim N(\mu_\ell, V_\ell)$
- based on acquired samples at previous iteration, we fit models (least squares)
  - $\mu^{\text{model}}_\ell = c_\alpha h^{\alpha}_\ell$
  - $V^{\text{model}}_\ell = c_\beta h^{\beta}_\ell$

- We take a Normal-Gamma prior for $(\mu_\ell, V_\ell)$, with mode in $(\mu^{\text{model}}_\ell, V^{\text{model}}_\ell)$
- Then $\hat{V}_\ell$ is the MAP Bayesian estimator based on the Normal-Gamma prior and the actual samples acquired at iteration $j$

Effectively, we have

$$M_\ell = 0 \quad \quad \hat{V}_\ell = V^{\text{model}}_\ell \quad \quad \text{(prior model)}$$

$$M_\ell \to \infty \quad \quad \hat{V}_\ell \approx V^{\text{MC}}_\ell \quad \quad \text{(sample variance)}$$

$\hat{V}_\ell$ is then used to determine the sample sizes $M_\ell$ for the next iteration.
Continuation Multilevel Monte Carlo


**Idea:** Solve the problem with decreasing tolerances $tol^{(0)} > tol^{(1)} > \ldots \geq tol$. Use collected samples on all levels to improve the estimate of $V_\ell$ and $|\mathbb{E}[Q - Q_\ell]|$.

Estimator $\hat{V}_\ell$ of $V_\ell = \text{Var}[\Delta Q_\ell]$ at iteration $j$: MAP Bayesian estimator

- we make the ansatz $\Delta Q_\ell \sim N(\mu_\ell, V_\ell)$
- based on acquired samples at previous iteration, we fit models (least squares)
  - $\mu_\ell\text{model} = c_\alpha h_\ell^\alpha$
  - $V_\ell\text{model} = c_\beta h_\ell^\beta$
- We take a Normal-Gamma prior for $(\mu_\ell, V_\ell)$, with mode in $(\mu_\ell\text{model}, V_\ell\text{model})$
- Then $\hat{V}_\ell$ is the MAP Bayesian estimator based on the Normal-Gamma prior and the actual samples acquired at iteration $j$

Effectively, we have

$$
\begin{align*}
M_\ell &= 0 & \hat{V}_\ell &= V_\ell\text{model} & \text{(prior model)} \\
M_\ell &\to \infty & \hat{V}_\ell &\approx V_\ell\text{MC} & \text{(sample variance)}
\end{align*}
$$

$\hat{V}_\ell$ is then used to determine the sample sizes $M_\ell$ for the next iteration.
## Computation of pressure coefficient for NACA 0012 / NASA SC(2)-0012 airfoils

<table>
<thead>
<tr>
<th>Name</th>
<th>Nominal value</th>
<th>Uncertainty</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_\infty$</td>
<td>$T_n = 288.15 \ [K]$</td>
<td>$\mathcal{T}N(T_n, 2%, 110%, 90%)$</td>
</tr>
<tr>
<td>$\rho_\infty$</td>
<td>$\rho_n = 101325 \ [N/m^2]$</td>
<td>$\mathcal{T}N(\rho_n, 2%, 110%, 90%)$</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>$\alpha_n = 1.25^\circ$</td>
<td>$\mathcal{T}N(\alpha_n, 1%, 110%, 90%)$</td>
</tr>
<tr>
<td>$M$</td>
<td>$M_n = 0.8$</td>
<td>$\mathcal{T}N(M_n, 2%, 110%, 90%)$</td>
</tr>
<tr>
<td>$R_p$</td>
<td>0.01458398</td>
<td>$\mathcal{T}N(R_{P_n}, 2.5%, 110%, 90%)$</td>
</tr>
<tr>
<td>$R_S$</td>
<td>0.01458398</td>
<td>$\mathcal{T}N(R_{S_n}, 2.5%, 110%, 90%)$</td>
</tr>
<tr>
<td>$X_P$</td>
<td>0.30049047</td>
<td>$\mathcal{T}N(X_{P_n}, 2.5%, 110%, 90%)$</td>
</tr>
<tr>
<td>$X_S$</td>
<td>0.30049047</td>
<td>$\mathcal{T}N(X_{S_n}, 2.5%, 110%, 90%)$</td>
</tr>
<tr>
<td>$Y_P$</td>
<td>$-0.05994286$</td>
<td>$\mathcal{T}N(Y_{P_n}, 2.5%, 110%, 90%)$</td>
</tr>
<tr>
<td>$Y_S$</td>
<td>0.05994286</td>
<td>$\mathcal{T}N(Y_{S_n}, 2.5%, 110%, 90%)$</td>
</tr>
<tr>
<td>$C_P$</td>
<td>0.44213792</td>
<td>$\mathcal{T}N(C_{P_n}, 2.5%, 110%, 90%)$</td>
</tr>
<tr>
<td>$C_S$</td>
<td>$-0.44213792$</td>
<td>$\mathcal{T}N(C_{S_n}, 2.5%, 110%, 90%)$</td>
</tr>
<tr>
<td>$\theta_P$</td>
<td>8.3763395</td>
<td>$\mathcal{T}N(\theta_{P_n}, 2.5%, 110%, 90%)$</td>
</tr>
<tr>
<td>$\theta_S$</td>
<td>$-8.3763395$</td>
<td>$\mathcal{T}N(\theta_{S_n}, 2.5%, 110%, 90%)$</td>
</tr>
</tbody>
</table>

---

**Diagram:**

- $R_p$, $R_S$, $C_P$, $C_S$, $\alpha_\infty$, $M_\infty$, $x$, $y$, $\theta_p$, $\theta_s$
Computation of pressure coefficient for NACA 0012 / NASA SC(2)-0012 airfoils

Inviscid model (Euler); SU2 solver (Stanford) [Pisaroni-Leyland-N., AIAA Aviation, 2016]

<table>
<thead>
<tr>
<th>LEVEL</th>
<th>Airfoil nodes</th>
<th>Cells</th>
<th>Avg. Real Computational Time [s] (CPU)</th>
</tr>
</thead>
<tbody>
<tr>
<td>L0</td>
<td>41</td>
<td>6943</td>
<td>12.4 (32)</td>
</tr>
<tr>
<td>L1</td>
<td>81</td>
<td>11115</td>
<td>20.9 (38)</td>
</tr>
<tr>
<td>L2</td>
<td>161</td>
<td>19385</td>
<td>26.9 (44)</td>
</tr>
<tr>
<td>L3</td>
<td>321</td>
<td>36251</td>
<td>71.1 (50)</td>
</tr>
<tr>
<td>L4</td>
<td>641</td>
<td>71477</td>
<td>231.15 (56)</td>
</tr>
<tr>
<td>L5</td>
<td>1281</td>
<td>145005</td>
<td>422.0 (64)</td>
</tr>
</tbody>
</table>
MLMC vs MC for aerodynamic inviscid problems

Computational Complexity of MC and MLMC

Levels and Samples per Level for $\varepsilon_r = 0.01$
Robustness of C-MLMC estimator

Variability over 10 repetitions of the C-MLMC algorithm for different parameters in the Normal-Gamma prior.
Outline

1 Motivating example

2 Multilevel Monte Carlo for expectations

3 MLMC for moments and distributions

4 Risk averse optimization with MLMC

5 Conclusions
Beyond expectations: computation of central moments

**Goal**: compute $\mu_p(Q) = \mathbb{E}[(Q - \mathbb{E}[Q])^p]$ 

How to apply and tune MLMC in this case? [Bierig-Chernov 2015-2016] use biased central moments estimators.

Alternatively, use $h$-statistics [Pisaroni-Krumscheid-N. 2017]. Given iid sample $\tilde{Q}_M = \{Q^{(1)}, \ldots, Q^{(M)}\}$,

$$h_p(\tilde{Q}_M): \text{unbiased estimator of } \mu_p(Q) \text{ with minimal variance}$$

**Multilevel estimator**: 

$$h_p^{MLMC} = \sum_{\ell=0}^{L} (h_p(\tilde{Q}_{\ell,M_\ell}) - h_p(\tilde{Q}_{\ell-1,M_\ell}))$$

with $\tilde{Q}_{\ell,M_\ell}, \tilde{Q}_{\ell-1,M_\ell}$ generated with the same noise (highly correlated)

**Mean squared error**: 

$$\text{MSE}(h_p^{MLMC}) = (\mu_p(Q) - \mu_p(Q_L))^2 + \sum_{\ell=0}^{L} \frac{V_{\ell,p}}{M_\ell}$$

where $V_{\ell,p} = M_\ell \text{Var}[h_p(\tilde{Q}_{\ell,M_\ell}) - h_p(\tilde{Q}_{\ell-1,M_\ell})]$.

Same “formal” structure as for expectation, but now we need to estimate $|\mu_p(Q) - \mu_p(Q_L)|$ and $V_{\ell,p}$ to tune the MLMC algorithm.
Beyond expectations: computation of central moments

**Goal:** compute $\mu_p(Q) = \mathbb{E}[(Q - \mathbb{E}[Q])^p]$

How to apply and tune MLMC in this case? [Bierig-Chernov 2015-2016] use biased central moments estimators.

Alternatively, use *h-statistics* [Pisaroni-Krumscheid-N. 2017]. Given iid sample $\vec{Q}_M = \{ Q^{(1)}, \ldots, Q^{(M)} \}$,

$$h_p(\vec{Q}_M) : \text{unbiased estimator of } \mu_p(Q) \text{ with minimal variance}$$

**Multilevel estimator:**

$$h_p^{MLMC} = \sum_{\ell=0}^{L} (h_p(\vec{Q}_{\ell,M_\ell}) - h_p(\vec{Q}_{\ell-1,M_\ell}))$$

with $(\vec{Q}_{\ell,M_\ell}, \vec{Q}_{\ell-1,M_\ell})$ generated with the same noise (highly correlated)

**Mean squared error:**

$$\text{MSE}(h_p^{MLMC}) = (\mu_p(Q) - \mu_p(Q_L))^2 + \sum_{\ell=0}^{L} \frac{V_{\ell,p}}{M_\ell}$$

where $V_{\ell,p} = M_\ell \text{Var}[h_p(\vec{Q}_{\ell,M_\ell}) - h_p(\vec{Q}_{\ell-1,M_\ell})]$.

Same “formal” structure as for expectation, but now we need to estimate $|\mu_p(Q) - \mu_p(Q_L)|$ and $V_{\ell,p}$ to tune the MLMC algorithm.
Beyond expectations: computation of central moments

**Goal:** compute $\mu_p(Q) = \mathbb{E}[(Q - \mathbb{E}[Q])^p]$

How to apply and tune MLMC in this case? [Bierig-Chernov 2015-2016] use biased central moments estimators.

Alternatively, use $h$-statistics [Pisaroni-Krumscheid-N. 2017]. Given iid sample $\mathbf{Q}_M = \{Q^{(1)}, \ldots, Q^{(M)}\}$,

$$h_p(\mathbf{Q}_M) : \text{unbiased estimator of } \mu_p(Q) \text{ with minimal variance}$$

**Multilevel estimator:**

$$h_p^{\text{MLMC}} = \sum_{\ell=0}^{L} (h_p(\mathbf{Q}_{\ell,M}) - h_p(\mathbf{Q}_{\ell-1,M}))$$

with $(\mathbf{Q}_{\ell,M}, \mathbf{Q}_{\ell-1,M})$ generated with the same noise (highly correlated)

**Mean squared error:**

$$\text{MSE}(h_p^{\text{MLMC}}) = (\mu_p(Q) - \mu_p(Q_L))^2 + \sum_{\ell=0}^{L} \frac{V_{\ell,p}}{M_\ell}$$

where $V_{\ell,p} = M_\ell \text{Var}[h_p(\mathbf{Q}_{\ell,M}) - h_p(\mathbf{Q}_{\ell-1,M})]$.

Same “formal” structure as for expectation, but now we need to estimate $|\mu_p(Q) - \mu_p(Q_L)|$ and $V_{\ell,p}$ to tune the MLMC algorithm.
Beyond expectations: computation of central moments

**Goal**: compute $\mu_p(Q) = \mathbb{E}[(Q - \mathbb{E}[Q])^p]$.

How to apply and tune MLMC in this case? [Bierig-Chernov 2015-2016] use biased central moments estimators.

Alternatively, use *h*-statistics [Pisaroni-Krumscheid-N. 2017]. Given iid sample $\tilde{Q}_M = \{Q^{(1)}, \ldots, Q^{(M)}\}$,

$$h_p(\tilde{Q}_M) : \text{unbiased estimator of } \mu_p(Q) \text{ with minimal variance}$$

**Multilevel estimator**:

$$h_p^{\text{MLMC}} = \sum_{\ell=0}^{L} (h_p(\tilde{Q}_{\ell,M_\ell}) - h_p(\tilde{Q}_{\ell-1,M_\ell}))$$

with $(\tilde{Q}_{\ell,M_\ell}, \tilde{Q}_{\ell-1,M_\ell})$ generated with the same noise (highly correlated).

**Mean squared error**:

$$\text{MSE}(h_p^{\text{MLMC}}) = (\mu_p(Q) - \mu_p(Q_L))^2 + \sum_{\ell=0}^{L} \frac{V_{\ell,p}}{M_\ell}$$

where $V_{\ell,p} = M_\ell \text{Var}[h_p(\tilde{Q}_{\ell,M_\ell}) - h_p(\tilde{Q}_{\ell-1,M_\ell})]$.

Same “formal” structure as for expectation, but now we need to estimate $|\mu_p(Q) - \mu_p(Q_L)|$ and $V_{\ell,p}$ to tune the MLMC algorithm.
Beyond expectations: computation of central moments

**Goal:** compute $\mu_p(Q) = \mathbb{E}[(Q - \mathbb{E}[Q])^p]$

How to apply and tune MLMC in this case? [Bierig-Chernov 2015-2016] use biased central moments estimators.

Alternatively, use $h$-statistics [Pisaroni-Krumscheid-N. 2017]. Given iid sample $\vec{Q}_M = \{Q^{(1)}, \ldots, Q^{(M)}\}$,

$$h_p(\vec{Q}_M) : \text{unbiased estimator of } \mu_p(Q) \text{ with minimal variance}$$

**Multilevel estimator:**

$$h_p^{\text{MLMC}} = \sum_{\ell=0}^{L} (h_p(\vec{Q}_\ell, M_\ell) - h_p(\vec{Q}_{\ell-1}, M_\ell))$$

with $(\vec{Q}_\ell, M_\ell, \vec{Q}_{\ell-1}, M_\ell)$ generated with the same noise (highly correlated)

**Mean squared error:**

$$\text{MSE}(h_p^{\text{MLMC}}) = (\mu_p(Q) - \mu_p(Q_L))^2 + \sum_{\ell=0}^{L} \frac{V_{\ell,p}}{M_\ell}$$

where $V_{\ell,p} = M_\ell \text{Var}[h_p(\vec{Q}_\ell, M_\ell) - h_p(\vec{Q}_{\ell-1}, M_\ell)]$.

Same “formal” structure as for expectation, but now we need to estimate $|\mu_p(Q) - \mu_p(Q_\ell)|$ and $V_{\ell,p}$ to tune the MLMC algorithm.
Beyond expectations: computation of central moments

**Complexity result** for $h_\ell = h_0 s^{-\ell}$

Assume $\mu_{2p}(Q_\ell) < \infty$ for all $\ell$ and there exist $\alpha, \beta, \gamma > 0$, $2\alpha \geq \min\{\beta, \gamma\}$ s.t.

- $|\mu_p(Q) - \mu_p(Q_\ell)| = O(h_\ell^\alpha)$,
- $V_{\ell,p} = O(h_\ell^\beta)$,
- $C_\ell = \text{Cost}(Q^{(i,\ell)}_\ell, Q^{(i,\ell)}_{\ell-1}) = O(h^{-\gamma})$,

Then, taking $L = O(tol^{1/\alpha})$ and $M_\ell = \left[ tol^{-2} \sqrt{\frac{V_{\ell,p}}{C_\ell}} \left( \sum_{k=0}^{L} \sqrt{C_k V_{k,p}} \right) \right]$ leads to

$$\text{MSE}(h_p^{\text{MLMC}}) \lesssim tol^2 \quad \text{and} \quad W(h_p^{\text{MLMC}}) \lesssim \begin{cases} tol^{-2}, & \beta > \gamma \\ tol^{-2} |\log(tol)|^2, & \beta = \gamma \\ tol^{-2} - \frac{\gamma - \beta}{\alpha}, & \beta < \gamma \end{cases}$$
Beyond expectations: computation of central moments

**Technical difficulty:** how to estimate the variances $V_{\ell,p}$

Define $\tilde{X}_{\ell,M\ell}^+ = \tilde{Q}_{\ell,M\ell} + \tilde{Q}_{\ell-1,M\ell}$, $\tilde{X}_{\ell,M\ell}^- = \tilde{Q}_{\ell,M\ell} - \tilde{Q}_{\ell-1,M\ell}$

$\Delta_\ell h_p = h_p(\tilde{Q}_{\ell,M\ell}) - h_p(\tilde{Q}_{\ell-1,M\ell})$ can be expressed as a power sum

$$\Delta_\ell h_p = \sum_{a+b \leq p} S_{a,b}(\tilde{X}_{\ell,M\ell}^+, \tilde{X}_{\ell,M\ell}^-), \quad S_{a,b}(\tilde{X}, \tilde{Y}) = \sum_i (X^{(i)})^a (Y^{(i)})^b$$

Unbiased estimators $\hat{V}_{\ell,p}$ of $V_{\ell,p}$ can be computed in closed form starting from the power terms $S_{a,b}(\tilde{X}_{\ell,M\ell}^+, \tilde{X}_{\ell,M\ell}^-)$ [Pisaroni-Krumscheid-N. 2017].
Beyond expectations: computation of central moments

**Technical difficulty**: how to estimate the variances $V_{\ell,p}$

Define

$$
\vec{X}^+_{\ell,M_\ell} = \vec{Q}_{\ell,M_\ell} + \vec{Q}_{\ell-1,M_\ell}, \quad \vec{X}^-_{\ell,M_\ell} = \vec{Q}_{\ell,M_\ell} - \vec{Q}_{\ell-1,M_\ell}
$$

$$
\Delta_\ell h_p = h_p(\vec{Q}_{\ell,M_\ell}) - h_p(\vec{Q}_{\ell-1,M_\ell}) \text{ can be expressed as a power sum}
$$

$$
\Delta_\ell h_p = \sum_{a+b \leq p} S_{a,b}(\vec{X}^+_{\ell,M_\ell}, \vec{X}^-_{\ell,M_\ell}), \quad S_{a,b}(\vec{X}^+, \vec{Y}) = \sum_i (X^{(i)})^a (Y^{(i)})^b
$$

Unbiased estimators $\hat{V}_{\ell,p}$ of $V_{\ell,p}$ can be computed in closed form starting from the power terms $S_{a,b}(\vec{X}^+_{\ell,M_\ell}, \vec{X}^-_{\ell,M_\ell})$ [Pisaroni-Krumscheid-N. 2017].
Beyond expectations: computation of central moments

**Technical difficulty**: how to estimate the variances $V_{\ell,p}$

Define

\[ \tilde{X}^+_{\ell,M_\ell} = \tilde{Q}_{\ell,M_\ell} + \tilde{Q}_{\ell-1,M_\ell}, \quad \tilde{X}^-_{\ell,M_\ell} = \tilde{Q}_{\ell,M_\ell} - \tilde{Q}_{\ell-1,M_\ell} \]

\[ \Delta_\ell h_p = h_p(\tilde{Q}_{\ell,M_\ell}) - h_p(\tilde{Q}_{\ell-1,M_\ell}) \]

can be expressed as a power sum

\[ \Delta_\ell h_p = \sum_{a+b \leq p} S_{a,b}(\tilde{X}^+_{\ell,M_\ell}, \tilde{X}^-_{\ell,M_\ell}), \quad S_{a,b}(\tilde{X}, \tilde{Y}) = \sum_i (X^{(i)})^a (Y^{(i)})^b \]

Unbiased estimators $\hat{V}_{\ell,p}$ of $V_{\ell,p}$ can be computed in closed form starting from the power terms $S_{a,b}(\tilde{X}^+_{\ell,M_\ell}, \tilde{X}^-_{\ell,M_\ell})$ [Pisaroni-Krumscheid-N. 2017].
Beyond expectations: char. function, CDF, and more

Some derived quantities can be written as parametric expectations

**Example 1**: Characteristic function of $Q$

$$
\Phi(\theta) = \mathbb{E}[\phi(\theta, Q)], \quad \phi(\theta, Q) = e^{i\theta Q}
$$

$\Rightarrow$ we can compute $\Phi(\theta_j)$ by MLMC on a set of points $\theta_j$.

**Example 2**: CDF of $Q$

$$
F(\theta) = \mathbb{E}[\phi(\theta, Q)], \quad \phi(\theta, Q) = 1_{\{Q \leq \theta\}}
$$

**Problem**: $\phi(\theta, Q)$ is not smooth! When applying MLMC, the variance of the differences, $V_\ell = \text{Var}[\phi(\theta, Q_\ell) - \phi(\theta, Q_{\ell-1})]$ will decay slowly. No much gain in MLMC.

**Remedies**:

- [Giles-Nagapetyan-Ritter 2015] smoothing: $F_\varepsilon(\theta) = \mathbb{E}[\phi_\varepsilon(\theta, Q)]$. Technical difficulty: $\varepsilon$ should depend on the required tolerance $\Rightarrow$ difficult tuning of MLMC
- [Bierig-Chernov 2016] approximate $F$ or pdf based on moments (see Alexey’s talk)
- [Krumscheid-N. 2017] anti-derivative approach: $F(\theta) = \Phi'(\theta)$ with $\Phi(\theta) = \mathbb{E}[\phi(\theta, Q)]$ and $\phi(\theta, \cdot)$ Lipschitz continuous.
Beyond expectations: char. function, CDF, and more

Some derived quantities can be written as parametric expectations

**Example 1**: Characteristic function of $Q$

$$\Phi(\theta) = \mathbb{E}[\phi(\theta, Q)], \quad \phi(\theta, Q) = e^{i\theta Q}$$

$\rightsquigarrow$ we can compute $\Phi(\theta_j)$ by MLMC on a set of points $\theta_j$.

**Example 2**: CDF of $Q$

$$F(\theta) = \mathbb{E}[\phi(\theta, Q)], \quad \phi(\theta, Q) = 1_{\{Q \leq \theta\}}$$

**Problem**: $\phi(\theta, Q)$ is not smooth! When applying MLMC, the variance of the differences, $V_\ell = \text{Var}[\phi(\theta, Q_\ell) - \phi(\theta, Q_{\ell-1})]$ will decay slowly. No much gain in MLMC.

**Remedies**:

- [Giles-Nagapetyan-Ritter 2015] smoothing: $F_\epsilon(\theta) = \mathbb{E}[\phi_\epsilon(\theta, Q)]$. Technical difficulty: $\epsilon$ should depend on the required tolerance $\rightsquigarrow$ difficult tuning of MLMC
- [Bierig-Chernov 2016] approximate $F$ or pdf based on moments (see Alexey’s talk)
- [Krbmscheid-N. 2017] anti-derivative approach: $F(\theta) = \Phi'(\theta)$ with $\Phi(\theta) = \mathbb{E}[\phi(\theta, Q)]$ and $\phi(\theta, \cdot)$ Lipschitz continuous.
Beyond expectations: char. function, CDF, and more

Some derived quantities can be written as parametric expectations

**Example 1**: Characteristic function of $Q$

$$\Phi(\theta) = \mathbb{E}[\phi(\theta, Q)], \quad \phi(\theta, Q) = e^{i\theta Q}$$

$\rightarrow$ we can compute $\Phi(\theta_j)$ by MLMC on a set of points $\theta_j$.

**Example 2**: CDF of $Q$

$$F(\theta) = \mathbb{E}[\phi(\theta, Q)], \quad \phi(\theta, Q) = 1\{Q \leq \theta\}$$

**Problem**: $\phi(\theta, Q)$ is not smooth! When applying MLMC, the variance of the differences, $V_\ell = \text{Var}[\phi(\theta, Q_\ell) - \phi(\theta, Q_{\ell-1})]$ will decay slowly. **No much gain in MLMC.**

**Remedies**:

- [Giles-Nagapetyan-Ritter 2015] smoothing: $F_\varepsilon(\theta) = \mathbb{E}[\phi_\varepsilon(\theta, Q)]$. Technical difficulty: $\varepsilon$ should depend on the required tolerance $\rightarrow$ difficult tuning of MLMC
- [Bierig-Chernov 2016] approximate $F$ or pdf based on moments (see Alexey’s talk)
- [Krumscheid-N. 2017] anti-derivative approach: $F(\theta) = \Phi'(\theta)$ with $\Phi(\theta) = \mathbb{E}[\phi(\theta, Q)]$ and $\phi(\theta, \cdot)$ Lipschitz continuous.
Beyond expectations: char. function, CDF, and more

Some derived quantities can be written as parametric expectations

**Example 1**: Characteristic function of $Q$

$$\Phi(\theta) = \mathbb{E}[\phi(\theta, Q)], \quad \phi(\theta, Q) = e^{i\theta Q}$$

$\Rightarrow$ we can compute $\Phi(\theta_j)$ by MLMC on a set of points $\theta_j$.

**Example 2**: CDF of $Q$

$$F(\theta) = \mathbb{E}[\phi(\theta, Q)], \quad \phi(\theta, Q) = \mathbb{1}_{\{Q \leq \theta\}}$$

**Problem**: $\phi(\theta, Q)$ is not smooth! When applying MLMC, the variance of the differences, $V_\ell = \text{Var}[\phi(\theta, Q_\ell) - \phi(\theta, Q_{\ell-1})]$ will decay slowly. No much gain in MLMC.

**Remedies**:

- [Giles-Nagapetyan-Ritter 2015] smoothing: $F_\epsilon(\theta) = \mathbb{E}[\phi_\epsilon(\theta, Q)]$. Technical difficulty: $\epsilon$ should depend on the required tolerance $\Rightarrow$ difficult tuning of MLMC
- [Bierig-Chernov 2016] approximate $F$ or pdf based on moments (see Alexey’s talk)
- [Krumschied-N. 2017] anti-derivative approach: $F(\theta) = \Phi'(\theta)$ with $\Phi(\theta) = \mathbb{E}[\phi(\theta, Q)]$ and $\phi(\theta, \cdot)$ Lipschitz continuous.
Anti-derivative approach to CDF computation

For any $\tau \in (0, 1)$ define

$$
\Phi_\tau(\theta) = \mathbb{E}[\phi_\tau(\theta, Q)], \quad \phi_\tau(\theta, Q) = \theta + \frac{1}{1 + \tau}(Q - \theta)_+
$$

Then

$$
F(\theta) = (1 - \tau)\Phi'_\tau(\theta) + \tau
$$

and MLMC can be effectively used to approximate $\Phi_\tau(\theta)$ and its derivatives.

Moreover, from the approximation of $\Phi_\tau$ and its derivatives we can get for free

- pdf: $p(\theta) = F'(\theta) = (1 - \tau)\Phi''_\tau(\theta)$
- $\tau$-quantile: $q_\tau = \inf\{\theta : F(\theta) \geq \tau\} = \arg\min_{\theta \in \mathbb{R}} \Phi_\tau(\theta)$
- Conditional Value at Risk

$$
CVaR_\tau = \frac{1}{1 - \tau} \int_{q_\tau}^{\infty} xdF(x) = \min_{\theta \in \mathbb{R}} \Phi_\tau(\theta)
$$
Anti-derivative approach to CDF computation

For any $\tau \in (0, 1)$ define

$$
\Phi_{\tau}(\theta) = \mathbb{E}[\phi_{\tau}(\theta, Q)], \quad \phi_{\tau}(\theta, Q) = \theta + \frac{1}{1 + \tau} (Q - \theta)_+
$$

Then

$$
F(\theta) = (1 - \tau)\Phi'_{\tau}(\theta) + \tau
$$

and MLMC can be effectively used to approximate $\Phi_{\tau}(\theta)$ and its derivatives.

Moreover, from the approximation of $\Phi_{\tau}$ and its derivatives we can get for free

- pdf: $p(\theta) = F'(\theta) = (1 - \tau)\Phi''_{\tau}(\theta)$
- $\tau$-quantile: $q_\tau = \inf\{\theta : F(\theta) \geq \tau\} = \arg\min_{\theta \in \mathbb{R}} \Phi_{\tau}(\theta)$
- Conditional Value at Risk

$$
CVaR_\tau = \frac{1}{1 - \tau} \int_{q_\tau}^{\infty} x dF(x) = \min_{\theta \in \mathbb{R}} \Phi_{\tau}(\theta)
$$
Computing parametric expectations by MLMC

**Goal**: given $\phi(\theta, Q)$, approximate $\Phi(\theta) = \mathbb{E}[\phi(\theta, Q)]$ and its derivatives uniformly in $\Theta$.

Interpolation approach:

- introduce a grid $\tilde{\xi} = \{\xi_1, \ldots, \xi_n\} \subset \Theta$
- compute $\Phi^{\text{MLMC}}_L(\xi_j), j = 1, \ldots, n$ by MLMC (same sample of $Q_\ell$ for every $\xi_j$)
- Interpolate values $\Phi^{\text{MLMC}}_L(\tilde{\xi}) = \{\Phi^{\text{MLMC}}_L(\xi_j)\}_{j=1}^n$

\[ \hat{\Phi}_L = \mathcal{I}_n(\Phi^{\text{MLMC}}_L(\tilde{\xi})) \]

e.g. by spline or polynomial interpolation

Assumptions on $\mathcal{I}_n$ (valid for spline interpolation)

- $\|f - \mathcal{I}_n(f(\tilde{\xi}))\|_{L^\infty(\Theta)} \leq c_1 n^{k+1}$, if $f \in C^{k+1}(\bar{\Theta})$
- $\|\mathcal{I}_n\tilde{x}\|_{L^\infty(\Theta)} \leq c_2 \|	ilde{x}\|_{\ell^\infty}$, $\forall \tilde{x} \in \mathbb{R}^n$
- \(\text{Cost}(\mathcal{I}_n(\tilde{x})) \leq c_3 n\)
Computing parametric expectations by MLMC

**Goal:** given $\phi(\theta, Q)$, approximate $\Phi(\theta) = \mathbb{E}[\phi(\theta, Q)]$ and its derivatives uniformly in $\Theta$.

Interpolation approach:
- introduce a grid $\vec{\xi} = \{\xi_1, \ldots, \xi_n\} \subset \Theta$
- compute $\Phi_{L}^{MLMC}(\xi_j), j = 1, \ldots, n$ by MLMC (same sample of $Q_\ell$ for every $\xi_j$)
- Interpolate values $\Phi_{L}^{MLMC}(\vec{\xi}) = \{\Phi_{L}^{MLMC}(\xi_j)\}_{j=1}^{n}$
  
  $$\hat{\Phi}_L = \mathcal{I}_n(\Phi_{L}^{MLMC}(\vec{\xi}))$$

  e.g. by spline or polynomial interpolation

Assumptions on $\mathcal{I}_n$ (valid for spline interpolation):
- $\|f - \mathcal{I}_n(f(\vec{\xi}))\|_{L^\infty(\Theta)} \leq c_1 n^{k+1}$, if $f \in C^{k+1}(\bar{\Theta})$
- $\|\mathcal{I}_n \vec{x}\|_{L^\infty(\Theta)} \leq c_2 \|\vec{x}\|_{\ell^\infty}$, $\forall \vec{x} \in \mathbb{R}^n$
- $\text{Cost}(\mathcal{I}_n(\vec{x})) \leq c_3 n$
Computing parametric expectations by MLMC

**Goal:** Given \( \phi(\theta, Q) \), approximate \( \Phi(\theta) = \mathbb{E}[\phi(\theta, Q)] \) and its derivatives uniformly in \( \Theta \).

**Interpolation approach:**
- Introduce a grid \( \vec{\xi} = \{\xi_1, \ldots, \xi_n\} \subset \Theta \)
- Compute \( \Phi_{MLMC}^L(\xi_j), j = 1, \ldots, n \) by MLMC (same sample of \( Q_\ell \) for every \( \xi_j \))
- Interpolate values \( \Phi_{MLMC}^L(\vec{\xi}) = \{\Phi_{MLMC}^L(\xi_j)\}_{j=1}^n \)
  \[ \hat{\Phi}_L = \mathcal{I}_n(\Phi_{MLMC}^L(\vec{\xi})) \]
  e.g. by spline or polynomial interpolation

**Assumptions on \( \mathcal{I}_n \) (valid for spline interpolation):**
- \( \|f - \mathcal{I}_n(f(\vec{\xi}))\|_{L^\infty(\Theta)} \leq c_1 n^{k+1}, \quad \text{if} \ f \in C^{k+1}(\vec{\Theta}) \)
- \( \|\mathcal{I}_n\vec{x}\|_{L^\infty(\Theta)} \leq c_2 \|\vec{x}\|_{\ell^\infty}, \quad \forall \vec{x} \in \mathbb{R}^n \)
- \( \text{Cost}(\mathcal{I}_n(\vec{x})) \leq c_3 n \)
Error splitting

Define the mean squared error:

\[ \text{MSE}(\hat{\Phi}_L) = \mathbb{E}[\| \Phi - \hat{\Phi}_L \|_{L^\infty(\Theta)}^2] \]

Notation: for \( \vec{x} \in \mathbb{R}^n \) define \( \text{Var}[\vec{x}] = \mathbb{E}[\| \vec{x} - \mathbb{E}[\vec{x}] \|_{l^\infty}^2] \)

Useful result: for \( \vec{x}^{(1)}, \ldots, \vec{x}^{(k)} \in \mathbb{R}^n \) independent,

\[ \text{Var}\left[ \sum_{i=1}^k \vec{x}^{(i)} \right] \leq c \log(n) \sum_{i=1}^k \text{Var}[\vec{x}^{(i)}] \]

Error splitting

\[ \text{MSE}(\hat{\Phi}_L) \leq 3\| \Phi - I_n \Phi \|_{L^\infty}^2 + 3\| I_n \Phi - I_n \Phi_L \|_{L^\infty}^2 + 3\mathbb{E}[\| I_n \Phi_L - I_n \Phi_L^{\text{MLMC}} \|_{L^\infty}^2] \]

\[ \lesssim \| \Phi - I_n \Phi(\vec{\xi}) \|_{L^\infty}^2 + \| \Phi(\vec{\xi}) - \Phi_L(\vec{\xi}) \|_{L^\infty}^2 + \log(n) \sum_{\ell=0}^L \frac{V_\ell}{M_\ell} \]

with \( V_\ell = \text{Var}[\phi(\vec{\xi}, Q_\ell) - \phi(\vec{\xi}, Q_{\ell-1})] \). All terms can be estimated in practice. Optimization of MLMC based on estimators \( \hat{V}_\ell \).

[Pisaroni-Krumscheid-N. in preparation]
Error splitting

Define the mean squared error: \( \text{MSE}(\hat{\Phi}_L) = \mathbb{E}[\| \Phi - \hat{\Phi}_L \|^2_{L_\infty(\Theta)}] \)

**Notation:** for \( \bar{x} \in \mathbb{R}^n \) define \( \text{Var}[\bar{x}] = \mathbb{E}[\| \bar{x} - \mathbb{E}[\bar{x}] \|^2_{\ell_\infty}] \)

**Useful result:** for \( \bar{x}^{(1)}, \ldots, \bar{x}^{(k)} \in \mathbb{R}^n \) independent,

\[
\text{Var}\left[ \sum_{i=1}^{k} \bar{x}^{(i)} \right] \leq c \log(n) \sum_{i=1}^{k} \text{Var}[\bar{x}^{(i)}]
\]

Error splitting

\[
\text{MSE}(\hat{\Phi}_L) \leq 3\| \Phi - \mathcal{I}_n \Phi \|^2_{\infty} + 3\| \mathcal{I}_n \Phi - \mathcal{I}_n \Phi_L \|^2_{\infty} + 3\mathbb{E}[\| \mathcal{I}_n \Phi_L - \mathcal{I}_n \Phi_L^{\text{MLMC}} \|^2_{\infty}]
\]

\[
\lesssim \| \Phi - \mathcal{I}_n \Phi(\bar{\xi}) \|^2_{\infty} + \| \Phi(\bar{\xi}) - \Phi_L(\bar{\xi}) \|^2_{\infty} + \log(n) \sum_{\ell=0}^{L} \frac{V_{\ell}}{M_{\ell}}
\]

with \( V_{\ell} = \text{Var}[\phi(\bar{\xi}, Q_{\ell}) - \phi(\bar{\xi}, Q_{\ell-1})] \). All terms can be estimated in practice. Optimization of MLMC based on estimators \( \hat{\nu}_{\ell} \).

[Pisaroni-Krumscheid-N. in preparation]
Complexity analysis

**Complexity result** for $h_\ell = h_0 s^{-\ell}$ [Krumscheid-N. 2017]

Assume

- $\| \Phi - \Phi_\ell \|_{L^\infty(\Theta)} \leq c_1 h_\ell^\alpha$,
- $\mathbb{E} \left( \| \phi(\cdot, Q_\ell) - \phi(\cdot, Q_{\ell-1}) \|_{L^\infty(\Theta)}^2 \right) \leq c_2 h_\ell^\beta$,
- cost to simulate one realization of $\phi(\theta, Q_\ell) \leq c_3 h_\ell^{-\gamma}$.

If $\Phi \in C^{k+1}(\Theta)$, there exists an estimator $\hat{\Phi}_L$ s.t.

$$W(\hat{\Phi}_L) \lesssim tol^{-\left(2 + \frac{1}{k+1}\right)}|\log(tol)| + |\log(tol)| \begin{cases} tol^{-2}, & \text{if } \beta > \gamma, \\ tol^{-2}|\log(tol)|^2, & \text{if } \beta = \gamma, \\ tol^{-\left(2 + \frac{\gamma - \beta}{\alpha}\right)}, & \text{if } \beta < \gamma, \end{cases}$$

The first term accounts for the cost of computing the spline interpolation. This is often negligible for heavy computational models. It can be removed by taking $n = n_\ell$ (different spline interpolant on each level).

Neglecting the first term, the complexity is essentially the same as for simple expectations, up to an extra log factor.
Complexity analysis

**Complexity result** for $h_\ell = h_0 s^{-\ell}$ [Krumscheid-N. 2017]

Assume

- $\| \Phi - \Phi_\ell \|_{L^\infty(\Theta)} \leq c_1 h_\ell^\alpha$,
- $\mathbb{E} \left( \| \phi(\cdot, Q_\ell) - \phi(\cdot, Q_{\ell-1}) \|_{L^\infty(\Theta)}^2 \right) \leq c_2 h_\ell^\beta$,
- cost to simulate one realization of $\phi(\theta, Q_\ell) \leq c_3 h_\ell^{-\gamma}$.

If $\Phi \in C^{k+1}(\Theta)$, there exists an estimator $\hat{\Phi}_L$ s.t. $\text{MSE}(\hat{\Phi}_L) = O(tol^2)$ and

$$W(\hat{\Phi}_L) \lesssim tol^{-(2+\frac{1}{k+1})} |\log(tol)| + |\log(tol)| \begin{cases} 
    tol^{-2}, & \text{if } \beta > \gamma, \\
    tol^{-2} |\log(tol)|^2, & \text{if } \beta = \gamma, \\
    tol^{-(2+\frac{\gamma-\beta}{\alpha})}, & \text{if } \beta < \gamma.
\end{cases}$$

The first term accounts for the cost of computing the spline interpolation. This is often negligible for heavy computational models. It can be removed by taking $n = n_\ell$ (different spline interpolant on each level).

Neglecting the first term, the complexity is essentially the same as for simple expectations, up to an extra log factor.
Complexity result for derivatives [Krumscheid-N. 2017]

If $\Phi \in C^{2k+2}(\Theta)$ and $m \leq 2k + 1$, there exists an estimator $\hat{\Phi}_L$ s.t.

$$
\mathbb{E}[\| \frac{d^m}{d\theta^m} \Phi - \frac{d^m}{d\theta^m} \hat{\Phi}_L \|_\infty^2] = \mathcal{O}(tol^2)
$$

and

$$
W(\hat{\Phi}_L) \lesssim |\log(tol)| \begin{cases} 
tol^{-2} \frac{2k+2}{2k+2-m}, & \text{if } \beta > \gamma, \\
tol^{-2} \frac{2k+2}{2k+2-m} |\log(tol)|^2, & \text{if } \beta = \gamma, \\
tol^{-(2+\frac{\gamma-\beta}{\alpha})} \frac{2k+2}{2k+2-m}, & \text{if } \beta < \gamma, 
\end{cases}
$$

(neglecting the cost of interpolation)

This result applies to the approximation of CDF, quantiles and CVaR with $m = 1$ and PDF with $m = 2$. 
An example: the characteristic function

- An SDE model to describe a European call option, where the asset follows
  \[ dS = rS \, dt + \sigma S \, dW, \quad S(0) = S_0, \]
- Quantity of interest is the discounted “payoff”:
  \[ Q := e^{-rT} \max(S(T) - K, 0) \]
- Approximate characteristic function of \( Q \):
  \[ \Phi(\theta) = \mathbb{E}(\cos(\theta Q)) + i \mathbb{E}(\sin(\theta Q)) \equiv \Phi_1(\theta) + i \Phi_2(\theta), \]
- Milstein scheme with \( h_\ell = 2^{-\ell} T; \Theta = [-1, 1], r = \frac{1}{20}, \sigma = \frac{1}{5}, T = 1, K = 10 = S_0. \]
An example: the characteristic function

- An SDE model to describe a European call option, where the asset follows
  \[ dS = rS \, dt + \sigma S \, dW \,, \quad S(0) = S_0 \,, \]
- Quantity of interest is the discounted “payoff”: \( Q := e^{-rT} \max(S(T) - K, 0) \)
- Approximate characteristic function of \( Q \):
  \[ \Phi(\theta) = \mathbb{E}(\cos(\theta Q)) + i \mathbb{E}(\sin(\theta Q)) \equiv \Phi_1(\theta) + i \Phi_2(\theta) \,, \]
- Milstein scheme with \( h_\ell = 2^{-\ell} T \); \( \Theta = [-1, 1] \), \( r = \frac{1}{20} \), \( \sigma = \frac{1}{5} \), \( T = 1 \), \( K = 10 = S_0 \).
NASA Common Research Model

NASA CRM: aircraft configuration equipped with a contemporary supercritical transonic wing and a fuselage that is representative of a wide-body commercial transport aircraft.

<table>
<thead>
<tr>
<th>Q.ty</th>
<th>Reference</th>
<th>Uncertainty</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M_\infty$</td>
<td>0.85</td>
<td>$\mathcal{B}(2, 2, 0.05, M_\infty - 0.025)$</td>
</tr>
<tr>
<td>$Re_c$</td>
<td>$5 \cdot 10^6$</td>
<td>—</td>
</tr>
<tr>
<td>$T_{ref}$</td>
<td>310.928 [K]</td>
<td>$\mathcal{B}(2, 2, 30, T_{ref} - 15)$</td>
</tr>
<tr>
<td>$C_L$</td>
<td>0.3, 0.4, 0.5, 0.55</td>
<td>—</td>
</tr>
</tbody>
</table>

Spalart-Allmaras turbulence model, hybrid unstructured grids.
NASA Common Research Model
NASA Common Research Model

Skin_friction

- 0.01
- 0.0091
- 0.0082
- 0.0073
- 0.0064
- 0.0055
- 0.0046
- 0.0037
- 0.0028
- 0.0019
- 0.001
- 0.0001
NASA Common Research Model

MLMC for moments and distributions
Outline

1 Motivating example
2 Multilevel Monte Carlo for expectations
3 MLMC for moments and distributions
4 Risk averse optimization with MLMC
5 Conclusions
Risk averse optimization

\[
\min_{x \in X} \mathcal{R}(Q(x)), \quad X: \text{feasible design space}
\]

\(\mathcal{R}\): risk measure

Examples

- \(\mathcal{R}(Q) = \mathbb{E}[Q]\) (mean-based risk)
- \(\mathcal{R}(Q) = \mathbb{E}[Q] \pm \alpha \text{std}[Q]\)
- \(\mathcal{R}(Q) = q_\alpha [Q]\) (\(\alpha\)-quantile)
- \(\mathcal{R}(Q) = \text{CVaR}_\alpha [Q]\)
Risk averse optimization

\[
\min_{x \in X} \mathcal{R}(Q(x)), \quad X: \text{feasible design space}
\]

\(\mathcal{R}\): risk measure

**Examples**

- \(\mathcal{R}(Q) = \mathbb{E}[Q]\) (mean-based risk)
- \(\mathcal{R}(Q) = \mathbb{E}[Q] \pm \alpha \text{std}[Q]\)
- \(\mathcal{R}(Q) = q_{\alpha}[Q]\) (\(\alpha\)-quantile)
- \(\mathcal{R}(Q) = \text{CVaR}_{\alpha}[Q]\)
Combining MLMC with CMA-ES

Optimization done by Covariance Matrix Adaptation Evolutionary Algorithm (CMA-ES)

For each individual at each generation, risk measure computed by MLMC.
Airfoil optimization under operating uncertainties

\[
\begin{align*}
\min_{x \in X} & \mathcal{R}[C_D(x)] \\
\text{s.t} & \quad C_L(x) = C_L^*, \quad \text{thickness constraint}
\end{align*}
\]

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Reference (r)</th>
<th>Uncertainty</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating parameters</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$C_L$</td>
<td>0.5</td>
<td>$\mathcal{B}(2, 2, 0.1, M_\infty - 0.05)$</td>
</tr>
<tr>
<td>$M_\infty$</td>
<td>0.75</td>
<td>$-$</td>
</tr>
<tr>
<td>$R_e_c$</td>
<td>$6.5 \cdot 10^6$</td>
<td>$-$</td>
</tr>
<tr>
<td>$p_\infty$ [Pa]</td>
<td>101325</td>
<td>$-$</td>
</tr>
<tr>
<td>$T_\infty$ [K]</td>
<td>288.5</td>
<td>$-$</td>
</tr>
</tbody>
</table>
Risk averse optimization with MLMC

Qualitative comparison

Model: steady state Euler + boundary layer equation (MSES software)
Deterministic versus Robust optimization
Multi-objective optimization under operating uncertainties

\[
P\min_{x \in X} \{\mu_{C_D}(x) + \sigma_{C_D}(x), -\mu_{C_L}(x) + \sigma_{C_L}(x)\} \quad \text{(Pareto front)}
\]

Uncertainties in Mach number and Angle of Attack.

Deterministic Optimized Airfoils

Certainty in Mach number and Angle of Attack.

Robust Optimized Airfoils
Outline

1 Motivating example
2 Multilevel Monte Carlo for expectations
3 MLMC for moments and distributions
4 Risk averse optimization with MLMC
5 Conclusions
Conclusions and outlook

- Multilevel Monte Carlo is a very powerful technique that can dramatically reduce the computational cost of a UQ analysis compared to plain MC.
- The tuning of MLMC requires adaptive algorithms and reliable error and variances estimators.
- We have presented a way to compute higher order moments as well as cdf, quantiles, CVaR with MLMC and properly tune the method.
- The methodology has been successfully applied to forward UQ propagation and robust optimization under uncertainty in compressible aerodynamics.
Thank you for your attention!
References

M. Pisaroni.

M. Pisaroni, S. Krumscheid, F. Nobile.

M. Pisaroni, S. Krumscheid, F. Nobile.
Quantifying uncertain system outputs via the multilevel Monte Carlo method Part 2: distribution and robustness measures, in preparation.

S. Krumscheid, F. Nobile.

M. Pisaroni, F. Nobile, P. Leyland.

M. Pisaroni, F. Nobile, P. Leyland.

M. Pisaroni, F. Nobile, P. Leyland.
