

Multilevel Monte Carlo methods

Mike Giles

Mathematical Institute, University of Oxford

International Congress of Mathematicians

Rio de Janeiro, August 3rd, 2018

With acknowledgements to many collaborators over the past 12 years

Objectives

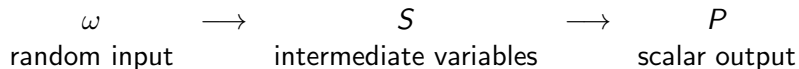
In presenting the multilevel Monte Carlo method, I hope to emphasise:

- the simplicity of the idea
- its flexibility – it's not prescriptive, more an approach
- there are lots of people working on a variety of applications

In doing this, I will focus on ideas rather than lots of numerical results.

Monte Carlo method

In stochastic models, we often have



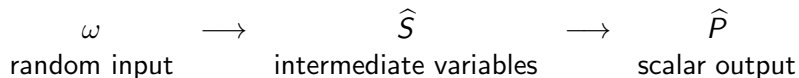
The Monte Carlo estimate for $\mathbb{E}[P]$ is an average of N independent samples $P(\omega^{(n)})$:

$$Y = N^{-1} \sum_{n=1}^N P(\omega^{(n)}).$$

This is unbiased, $\mathbb{E}[Y] = \mathbb{E}[P]$, and the Central Limit Theorem proves that as $N \rightarrow \infty$ the error becomes Normally distributed with variance $N^{-1}\mathbb{V}[P]$ so need $N = O(\varepsilon^{-2})$ samples to achieve ε RMS accuracy.

Monte Carlo method

In many cases, this is modified to



where \hat{S}, \hat{P} are approximations to S, P , in which case the MC estimate

$$\hat{Y} = N^{-1} \sum_{n=1}^N \hat{P}(\omega^{(n)})$$

is biased, and the Mean Square Error is

$$\mathbb{E}[(\hat{Y} - \mathbb{E}[P])^2] = N^{-1} \mathbb{V}[\hat{P}] + (\mathbb{E}[\hat{P}] - \mathbb{E}[P])^2$$

Greater accuracy requires larger N and smaller weak error $\mathbb{E}[\hat{P}] - \mathbb{E}[P]$.

SDE Path Simulation

My original interest was in SDEs (stochastic differential equations) for finance, which in a simple scalar case has the form

$$dS_t = a(S_t, t) dt + b(S_t, t) dW_t$$

where dW_t is the increment of a Brownian motion – Normally distributed with variance dt .

This is usually approximated by the simple Euler-Maruyama method

$$\widehat{S}_{t_{n+1}} = \widehat{S}_{t_n} + a(\widehat{S}_{t_n}, t_n) h + b(\widehat{S}_{t_n}, t_n) \Delta W_n$$

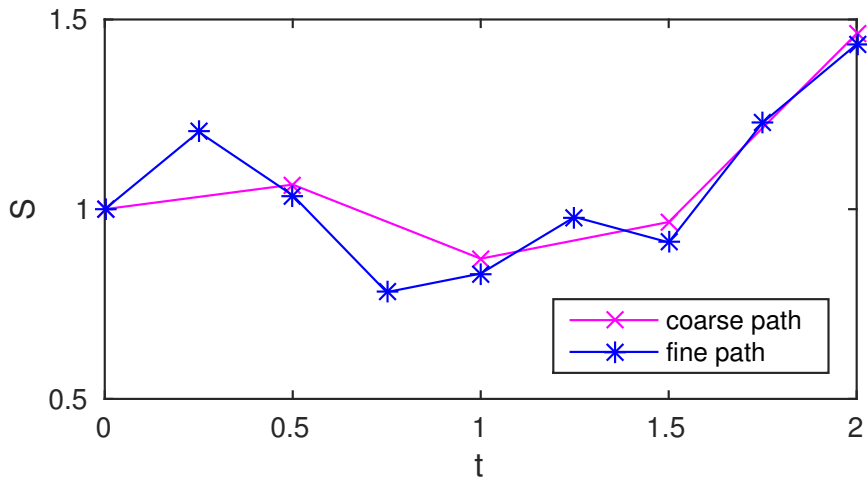
with uniform timestep h , and increments ΔW_n with variance h .

In simple applications, the output of interest is a function of the final value:

$$\widehat{P} \equiv f(\widehat{S}_T)$$

SDE Path Simulation

Geometric Brownian Motion: $dS_t = r S_t dt + \sigma S_t dW_t$



SDE Path Simulation

Two kinds of discretisation error:

Weak error:

$$\mathbb{E}[\widehat{P}] - \mathbb{E}[P] = O(h)$$

Strong error:

$$\left(\mathbb{E} \left[\sup_{[0, T]} (\widehat{S}_t - S_t)^2 \right] \right)^{1/2} = O(h^{1/2})$$

For reasons which will become clear, I prefer to use the Milstein discretisation for which the weak and strong errors are both $O(h)$.

SDE Path Simulation

The Mean Square Error is

$$N^{-1} \mathbb{V}[\hat{P}] + \left(\mathbb{E}[\hat{P}] - \mathbb{E}[P] \right)^2 \approx a N^{-1} + b h^2$$

If we want this to be ε^2 , then we need

$$N = O(\varepsilon^{-2}), \quad h = O(\varepsilon)$$

so the total computational cost is $O(\varepsilon^{-3})$.

To improve this cost we need to

- reduce N – variance reduction or Quasi-Monte Carlo methods
- reduce the cost of each path (on average) – MLMC

Two-level Monte Carlo

If we want to estimate $\mathbb{E}[\widehat{P}_1]$ but it is much cheaper to simulate $\widehat{P}_0 \approx \widehat{P}_1$, then since

$$\mathbb{E}[\widehat{P}_1] = \mathbb{E}[\widehat{P}_0] + \mathbb{E}[\widehat{P}_1 - \widehat{P}_0]$$

we can use the estimator

$$N_0^{-1} \sum_{n=1}^{N_0} \widehat{P}_0^{(0,n)} + N_1^{-1} \sum_{n=1}^{N_1} \left(\widehat{P}_1^{(1,n)} - \widehat{P}_0^{(1,n)} \right)$$

Benefit: if $\widehat{P}_1 - \widehat{P}_0$ is small, its variance will be small, so won't need many samples to accurately estimate $\mathbb{E}[\widehat{P}_1 - \widehat{P}_0]$, so cost will be reduced greatly.

Multilevel Monte Carlo

Natural generalisation: given a sequence $\hat{P}_0, \hat{P}_1, \dots, \hat{P}_L$

$$\mathbb{E}[\hat{P}_L] = \mathbb{E}[\hat{P}_0] + \sum_{\ell=1}^L \mathbb{E}[\hat{P}_\ell - \hat{P}_{\ell-1}]$$

we can use the estimator

$$N_0^{-1} \sum_{n=1}^{N_0} \hat{P}_0^{(0,n)} + \sum_{\ell=1}^L \left\{ N_\ell^{-1} \sum_{n=1}^{N_\ell} \left(\hat{P}_\ell^{(\ell,n)} - \hat{P}_{\ell-1}^{(\ell,n)} \right) \right\}$$

with independent estimation for each level of correction

Multilevel Monte Carlo

If we define

- C_0, V_0 to be cost and variance of \widehat{P}_0
- C_ℓ, V_ℓ to be cost and variance of $\widehat{P}_\ell - \widehat{P}_{\ell-1}$

then the total cost is $\sum_{\ell=0}^L N_\ell C_\ell$ and the variance is $\sum_{\ell=0}^L N_\ell^{-1} V_\ell$.

Using a Lagrange multiplier μ^2 to minimise the cost for a fixed variance

$$\frac{\partial}{\partial N_\ell} \sum_{k=0}^L (N_k C_k + \mu^2 N_k^{-1} V_k) = 0$$

gives

$$N_\ell = \mu \sqrt{V_\ell / C_\ell} \quad \implies \quad N_\ell C_\ell = \mu \sqrt{V_\ell C_\ell}$$

Multilevel Monte Carlo

Setting the total variance equal to ε^2 gives

$$\mu = \varepsilon^{-2} \left(\sum_{\ell=0}^L \sqrt{V_\ell C_\ell} \right)$$

and hence, the total cost is

$$\sum_{\ell=0}^L N_\ell C_\ell = \varepsilon^{-2} \left(\sum_{\ell=0}^L \sqrt{V_\ell C_\ell} \right)^2$$

in contrast to the standard cost which is approximately $\varepsilon^{-2} V_0 C_L$.

The MLMC cost savings are therefore approximately:

- V_L/V_0 , if $\sqrt{V_\ell C_\ell}$ increases with level
- C_0/C_L , if $\sqrt{V_\ell C_\ell}$ decreases with level

Multilevel Path Simulation

With SDEs, level ℓ corresponds to approximation using M^ℓ timesteps, giving approximate payoff \widehat{P}_ℓ at cost $C_\ell = O(h_\ell^{-1})$.

Simplest estimator for $\mathbb{E}[\widehat{P}_\ell - \widehat{P}_{\ell-1}]$ for $\ell > 0$ is

$$\widehat{Y}_\ell = N_\ell^{-1} \sum_{n=1}^{N_\ell} \left(\widehat{P}_\ell^{(n)} - \widehat{P}_{\ell-1}^{(n)} \right)$$

using same driving Brownian path for both levels.

Analysis gives $\text{MSE} = \sum_{\ell=0}^L N_\ell^{-1} V_\ell + \left(\mathbb{E}[\widehat{P}_L] - \mathbb{E}[P] \right)^2$

To make RMS error less than ε

- choose $N_\ell \propto \sqrt{V_\ell / C_\ell}$ so total variance is less than $\frac{1}{2} \varepsilon^2$
- choose L so that $\left(\mathbb{E}[\widehat{P}_L] - \mathbb{E}[P] \right)^2 < \frac{1}{2} \varepsilon^2$

Multilevel Path Simulation

For Lipschitz payoff functions $P \equiv f(S_T)$, we have

$$\begin{aligned} V_\ell &\equiv \mathbb{V} \left[\widehat{P}_\ell - \widehat{P}_{\ell-1} \right] \leq \mathbb{E} \left[(\widehat{P}_\ell - \widehat{P}_{\ell-1})^2 \right] \\ &\leq K^2 \mathbb{E} \left[(\widehat{S}_{T,\ell} - \widehat{S}_{T,\ell-1})^2 \right] \\ &= \begin{cases} O(h_\ell), & \text{Euler-Maruyama} \\ O(h_\ell^2), & \text{Milstein} \end{cases} \end{aligned}$$

and hence

$$V_\ell C_\ell = \begin{cases} O(1), & \text{Euler-Maruyama} \\ O(h_\ell), & \text{Milstein} \end{cases}$$

MLMC Theorem

(Slight generalisation of version in 2008 *Operations Research* paper)

If there exist independent estimators \hat{Y}_ℓ based on N_ℓ Monte Carlo samples, each costing C_ℓ , and positive constants $\alpha, \beta, \gamma, c_1, c_2, c_3$ such that $\alpha \geq \frac{1}{2} \min(\beta, \gamma)$ and

$$\text{i) } \left| \mathbb{E}[\hat{P}_\ell - P] \right| \leq c_1 2^{-\alpha \ell}$$

$$\text{ii) } \mathbb{E}[\hat{Y}_\ell] = \begin{cases} \mathbb{E}[\hat{P}_0], & \ell = 0 \\ \mathbb{E}[\hat{P}_\ell - \hat{P}_{\ell-1}], & \ell > 0 \end{cases}$$

$$\text{iii) } \mathbb{V}[\hat{Y}_\ell] \leq c_2 N_\ell^{-1} 2^{-\beta \ell}$$

$$\text{iv) } \mathbb{E}[C_\ell] \leq c_3 2^{\gamma \ell}$$

MLMC Theorem

then there exists a positive constant c_4 such that for any $\varepsilon < 1$ there exist L and N_ℓ for which the multilevel estimator

$$\hat{Y} = \sum_{\ell=0}^L \hat{Y}_\ell,$$

has a mean-square-error with bound $\mathbb{E} \left[\left(\hat{Y} - \mathbb{E}[P] \right)^2 \right] < \varepsilon^2$

with an expected computational cost C with bound

$$C \leq \begin{cases} c_4 \varepsilon^{-2}, & \beta > \gamma, \\ c_4 \varepsilon^{-2} (\log \varepsilon)^2, & \beta = \gamma, \\ c_4 \varepsilon^{-2 - (\gamma - \beta)/\alpha}, & 0 < \beta < \gamma. \end{cases}$$

MLMC Theorem

Two observations of optimality:

- MC simulation needs $O(\varepsilon^{-2})$ samples to achieve RMS accuracy ε .
When $\beta > \gamma$, the cost is optimal — $O(1)$ cost per sample on average.
(Would need multilevel QMC to further reduce costs)
- When $\beta < \gamma$, another interesting case is when $\beta = 2\alpha$, which corresponds to $\mathbb{E}[\widehat{Y}_\ell]$ and $\sqrt{\mathbb{E}[\widehat{Y}_\ell^2]}$ being of the same order as $\ell \rightarrow \infty$.
In this case, the total cost is $O(\varepsilon^{-\gamma/\alpha})$, which is the cost of a single sample on the finest level — again optimal.

MLMC generalisation

The theorem is for scalar outputs P , but it can be generalised to multi-dimensional (or infinite-dimensional) outputs with

$$\text{i) } \left\| \mathbb{E}[\widehat{P}_\ell - P] \right\| \leq c_1 2^{-\alpha \ell}$$

$$\text{ii) } \mathbb{E}[\widehat{Y}_\ell] = \begin{cases} \mathbb{E}[\widehat{P}_0], & \ell = 0 \\ \mathbb{E}[\widehat{P}_\ell - \widehat{P}_{\ell-1}], & \ell > 0 \end{cases}$$

$$\text{iii) } \mathbb{V}[\widehat{Y}_\ell] \equiv \mathbb{E} \left[\left\| \widehat{Y}_\ell - \mathbb{E}[\widehat{Y}_\ell] \right\|^2 \right] \leq c_2 N_\ell^{-1} 2^{-\beta \ell}$$

Original multilevel research by Heinrich in 1999 did this for parametric integration, estimating $g(\lambda) \equiv \mathbb{E}[f(x, \lambda)]$ for a finite-dimensional r.v. x .

MLMC work on SDEs

- Milstein discretisation for path-dependent options – G (2008)
- numerical analysis – G, Higham, Mao (2009), Avikainen (2009), G, Debrabant, Rößler (2012)
- financial sensitivities (“Greeks”) – Burgos (2011)
- jump-diffusion models – Xia (2011)
- Lévy processes – Dereich (2010), Marxen (2010), Dereich & Heidenreich (2011), Xia (2013), Kyprianou (2014)
- American options – Belomestny & Schoenmakers (2011)
- Milstein in higher dimensions without Lévy areas – G, Szpruch (2014)
- adaptive timesteps – Hoel, von Schwerin, Szepessy, Tempone (2012), G, Lester, Whittle (2014)

SPDEs

- quite natural application, with better cost savings than SDEs due to higher dimensionality
- range of applications

- ▶ Graubner & Ritter (Darmstadt) – parabolic
- ▶ G, Reisinger (Oxford) – parabolic (credit derivative application)

$$dp = -\mu \frac{\partial p}{\partial x} dt + \frac{1}{2} \frac{\partial^2 p}{\partial x^2} dt + \sqrt{\rho} \frac{\partial p}{\partial x} dW$$

with absorbing boundary $p(0, t) = 0$

- ▶ Cliffe, G, Scheichl, Teckentrup (Bath/Nottingham) – elliptic

$$\nabla \cdot (\kappa(\omega, x) \nabla p) = 0$$

where $\log \kappa(\omega, x)$ is a Gaussian field – Normally distributed at each point, and with a certain spatial correlation

- ▶ Barth, Jenny, Lang, Meyer, Mishra, Müller, Schwab, Sukys, Zollinger (ETH Zürich) – elliptic, parabolic, hyperbolic
- ▶ Harbrecht, Peters (Basel) – elliptic
- ▶ Efendiev (Texas A&M) – numerical homogenization
- ▶ Heitzinger (TU Vienna) – elliptic drift-diffusion-Poisson system

Engineering Uncertainty Quantification

Simplest possible example:

- 3D elliptic PDE, with uncertain boundary data
- grid spacing proportional to $2^{-\ell}$ on level ℓ
- cost is $O(2^{+3\ell})$, if using an efficient multigrid solver
- 2nd order accuracy means that

$$\begin{aligned}\widehat{P}_\ell(\omega) - P(\omega) &\approx c(\omega) 2^{-2\ell} \\ \implies \widehat{P}_{\ell-1}(\omega) - \widehat{P}_\ell(\omega) &\approx 3c(\omega) 2^{-2\ell}\end{aligned}$$

- hence, $\alpha=2$, $\beta=4$, $\gamma=3$
- cost is $O(\varepsilon^{-2})$ to obtain ε RMS accuracy
- this compares to $O(\varepsilon^{-3/2})$ cost for one sample on finest level, so $O(\varepsilon^{-7/2})$ for standard Monte Carlo

Non-geometric multilevel

Almost all applications of multilevel in the literature so far use a geometric sequence of levels, refining the timestep (or the spatial discretisation for PDEs) by a constant factor when going from level ℓ to level $\ell + 1$.

Coming from a multigrid background, this is very natural, but it is **NOT** a requirement of the multilevel Monte Carlo approach.

All MLMC needs is a sequence of levels with

- increasing accuracy
- increasing cost
- increasingly small difference between outputs on successive levels

Reduced Basis PDE approximation

Vidal-Codina, Nguyen, G, Peraire (2014) take a fine FE discretisation:

$$A(\omega) u = f(\omega)$$

and use a reduced basis approximation

$$u \approx \sum_{k=1}^K v_k u_k$$

to obtain a low-dimensional reduced system

$$A_r(\omega) v = f_r(\omega)$$

- larger $K \implies$ greater accuracy at greater cost
- in multilevel treatment, K_ℓ varies with level
- brute force optimisation determines the optimal number of levels, and reduced basis size on each level

Nested simulation

Nested simulation is interested in the estimation of

$$\mathbb{E} \left[g \left(\mathbb{E}[f(X, Y) | X] \right) \right]$$

for independent random variables X, Y .

If each individual $f(X, Y)$ can be sampled at unit cost then an MLMC treatment can use 2^ℓ samples on level ℓ .

For given sample X , a good “antithetic” estimator is

$$Z_\ell = g(\bar{f}) - \frac{1}{2} \left(g(\bar{f}^{(a)}) + g(\bar{f}^{(b)}) \right)$$

where

- $\bar{f}^{(a)}$ is an average of $f(X, Y)$ over $2^{\ell-1}$ independent samples for Y ;
- $\bar{f}^{(b)}$ is an average over a second independent set of $2^{\ell-1}$ samples;
- \bar{f} is an average over the combined set of 2^ℓ inner samples.

Nested simulation

Note that

$$\begin{aligned}\bar{f} &= \frac{1}{2} \left(\bar{f}^{(a)} + \bar{f}^{(b)} \right), \\ \implies \bar{f}^{(a)} &= \bar{f} + \frac{1}{2} \left(\bar{f}^{(a)} - \bar{f}^{(b)} \right), \\ \bar{f}^{(b)} &= \bar{f} - \frac{1}{2} \left(\bar{f}^{(a)} - \bar{f}^{(b)} \right).\end{aligned}$$

Doing a Taylor series expansion about \bar{f} then gives

$$Z_\ell \approx \frac{1}{2} g''(\bar{f}) \left(\bar{f}^{(a)} - \bar{f}^{(b)} \right)^2 = O(2^{-\ell})$$

which gives $\alpha = 1, \beta = 2, \gamma = 1$, and hence an $O(\varepsilon^{-2})$ complexity.

This has been used for pedestrian “flow” by Haji-Ali (2012) and credit modelling by Bujok, Hambly & Reisinger (2015).

Mixed precision computing

As more examples of the flexibility of the MLMC approach, the levels can correspond to different levels of computing precision

- $2\ell+2$ bits of precision on level ℓ when using FPGAs (Korn, Ritter, Wehn, 2014)
- IEEE half-precision on level 0, IEEE single precision on level 1, etc., when computing on CPUs or GPUs

or the different levels can use different random number generators

- level 0: 10-bit uniform random numbers, with table lookup to convert to approximate Normals
- level 1: 32-bit uniform random numbers, and more complex calculation of $\Phi^{-1}(U)$ to obtain Normals

Other MLMC applications

- parametric integration, integral equations (Heinrich, 1998)
- multilevel QMC (G, Waterhouse 2009, Dick, Kuo, Scheichl, Schwab, Sloan, 2014-18)
- stochastic chemical reactions (Anderson & Higham, 2012, 2014, Moraes & Tempone, 2016-17)
- MLMC for MCMC (Schwab & Stuart, 2013; Scheichl & Teckentrup, 2015)
- Coulomb collisions in plasma (Caflisch *et al*, 2013)
- invariant distribution of contractive Markov process (Glynn & Rhee, 2014)
- invariant distribution of contractive SDEs (G, Lester & Whittle, 2015)
- MLMC for rare events and reliability calculations (Ullmann, Papaioannou, 2015; Aslett, Nagapetyan, Vollmer, 2017)

Three MLMC extensions

- unbiased estimation – Rhee & Glynn (2015)
 - ▶ randomly selects the level for each sample
 - ▶ no bias, and finite expected cost and variance if $\beta > \gamma$
- Richardson-Romberg extrapolation – Lemaire & Pagès (2017)
 - ▶ reduces the weak error, and hence the number of levels required
 - ▶ particularly helpful when $\beta < \gamma$
- Multi-Index Monte Carlo – Haji-Ali, Nobile, Tempone (2015)
 - ▶ important extension to MLMC approach, combining MLMC with sparse grid methods (combination technique)

Randomised Multilevel Monte Carlo

Rhee & Glynn (2015) started from

$$\mathbb{E}[P] = \sum_{\ell=0}^{\infty} \mathbb{E}[\Delta P_{\ell}] = \sum_{\ell=0}^{\infty} p_{\ell} \mathbb{E}[\Delta P_{\ell}/p_{\ell}],$$

to develop an unbiased single-term estimator

$$Y = \Delta P_{\ell'} / p_{\ell'},$$

where ℓ' is a random index which takes value ℓ with probability p_{ℓ} .

$\beta > \gamma$ is required to simultaneously obtain finite variance and finite expected cost using

$$p_{\ell} \propto 2^{-(\beta+\gamma)\ell/2}.$$

The complexity is then $O(\varepsilon^{-2})$.

Multi-Index Monte Carlo

Standard “1D” MLMC truncates the telescoping sum

$$\mathbb{E}[P] = \sum_{\ell=0}^{\infty} \mathbb{E}[\Delta \hat{P}_{\ell}]$$

where $\Delta \hat{P}_{\ell} \equiv \hat{P}_{\ell} - \hat{P}_{\ell-1}$, with $\hat{P}_{-1} \equiv 0$.

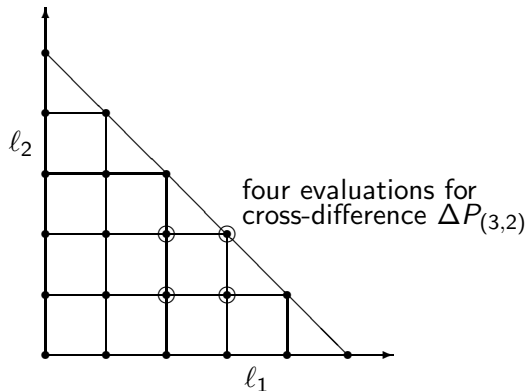
In “2D”, MIMC truncates the telescoping sum

$$\mathbb{E}[P] = \sum_{\ell_1=0}^{\infty} \sum_{\ell_2=0}^{\infty} \mathbb{E}[\Delta \hat{P}_{\ell_1, \ell_2}]$$

where $\Delta \hat{P}_{\ell_1, \ell_2} \equiv (\hat{P}_{\ell_1, \ell_2} - \hat{P}_{\ell_1-1, \ell_2}) - (\hat{P}_{\ell_1, \ell_2-1} - \hat{P}_{\ell_1-1, \ell_2-1})$

Different aspects of the discretisation vary in each “dimension” – for a 2D PDE, could use grid spacing $2^{-\ell_1}$ in direction 1, $2^{-\ell_2}$ in direction 2

Multi-Index Monte Carlo



MIMC truncates the summation in a way which minimises the cost to achieve a target MSE – quite similar to sparse grids.

Can achieve $O(\varepsilon^{-2})$ complexity for a wider range of SPDE and other applications than plain MLMC.

Conclusions

- multilevel idea is very simple; key question is how to apply it in new situations, and how to carry out the numerical analysis
- discontinuous output functions can cause problems, but there is a lot of experience now in coping with this
- there are also “tricks” which can be used in situations with poor strong convergence
- being used for an increasingly wide range of applications; biggest computational savings when coarsest (reasonable) approximation is much cheaper than finest
- currently, getting at least $100\times$ savings for SPDEs and stochastic chemical reaction simulations

References

Webpages for my research papers and talks:

`people.maths.ox.ac.uk/gilesm/mlmc.html`

`people.maths.ox.ac.uk/gilesm/slides.html`

Webpage for 70-page *Acta Numerica* review and MATLAB test codes:

`people.maths.ox.ac.uk/gilesm/acta/`

– contains references to almost all MLMC research up to 2015

Webpage for MLMC research community:

`people.maths.ox.ac.uk/gilesm/mlmc_community.html`