An introduction to Multilevel Monte Carlo methods

Mike Giles

Mathematical Institute, University of Oxford

UNSW, April 18, 2023

Acknowledgements to many collaborators:

Frances Kuo, Ian Sloan (UNSW), Rob Scheichl (Heidelberg),
Des Higham, Lukas Szpruch, Aretha Teckentrup (Edinburgh),
Al Haji-Ali (Heriot-Watt), Klaus Ritter (Kaiserslautern),
Takashi Goda (Tokyo), Matteo Croci (UT Austin),
Patrick Farrell, Ben Hambly, Christoph Reisinger (Oxford), ...

Objectives

In presenting the multilevel Monte Carlo method, I want 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

$$\underset{\text{random input}}{\omega} \longrightarrow \underset{\text{intermediate variables}}{S} \longrightarrow \underset{\text{scalar output}}{P}$$

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

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

This is unbiased, $\mathbb{E}[Y] = \mathbb{E}[P]$, and the Central Limit Theorem proves that as $N \to \infty$ the error becomes Normally distributed with variance $N^{-1}\mathbb{V}[P]$.

Monte Carlo method

In many cases, this is modified to

$$\underset{\text{random input}}{\omega} \longrightarrow \underset{\text{intermediate variables}}{\widehat{S}} \longrightarrow \underset{\text{scalar output}}{\widehat{P}}$$

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

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

is biased, and the Mean Square Error is

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

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

SDE Path Simulation

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

$$\mathrm{d}S_t = a(S_t, t)\ \mathrm{d}t + b(S_t, t)\,\mathrm{d}W_t$$

Here $\mathrm{d}W_t$ is the increment of a Brownian motion – Normally distributed with variance $\mathrm{d}t$.

This is usually approximated by the simple Euler-Maruyama method

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

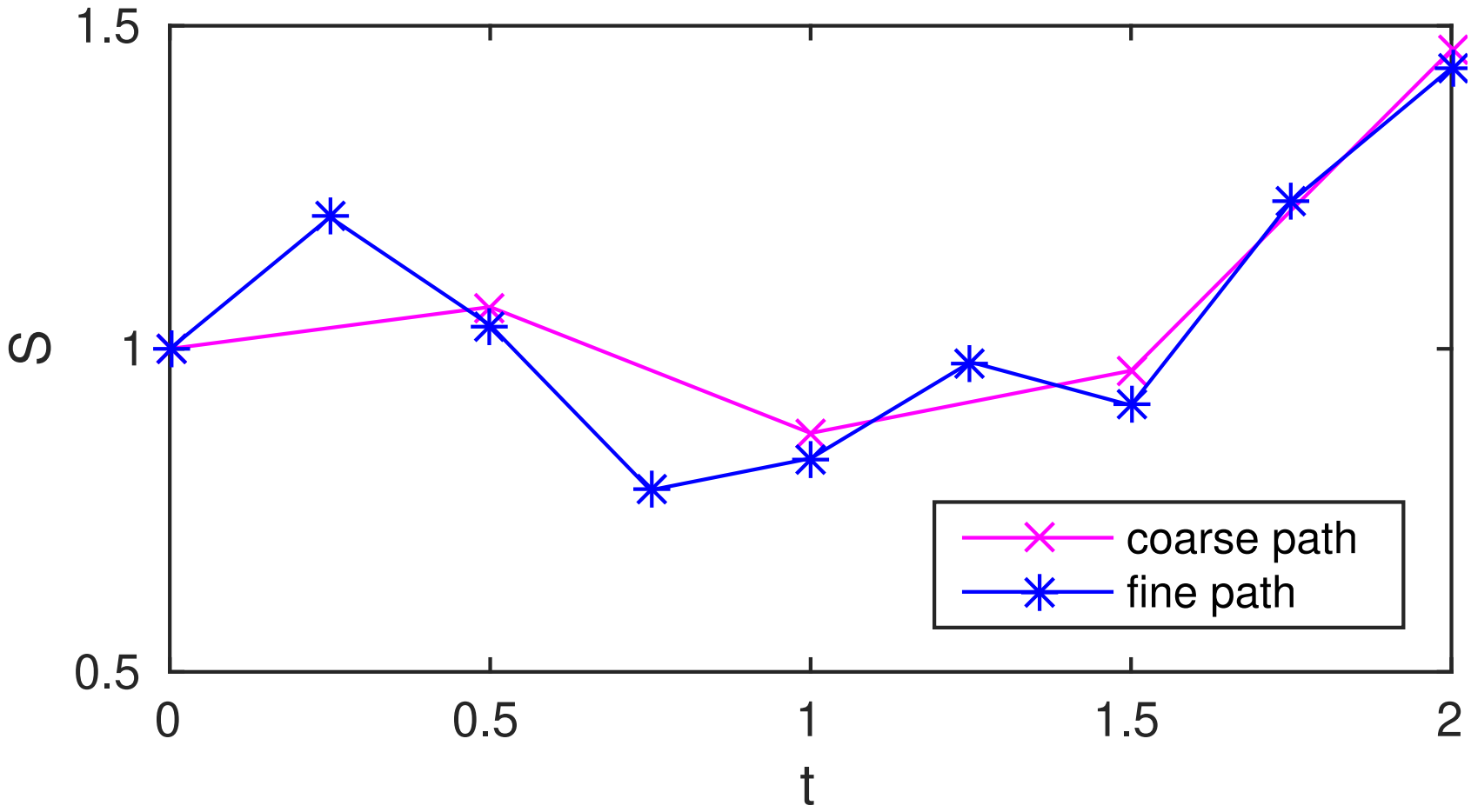
with uniform timestep h, and increments ΔW_n with variance h.

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: $\mathrm{d}S_t = r\,S_t\,\mathrm{d}t + \sigma\,S_t\,\mathrm{d}W_t$



SDE Path Simulation

Two kinds of discretisation error:

Weak error:

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

Strong error:

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

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

$$N^{-1}\,\mathbb{V}[\widehat{P}] + \Big(\mathbb{E}[\widehat{P}] - \mathbb{E}[P]\Big)^2 \;\approx\; a\,N^{-1} + b\,h^2$$

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

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

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.

Two-level Monte Carlo

Very similar to control variate variance reduction in which

- we want to estimate $\mathbb{E}[f]$
- there's some other output g for which we know $\mathbb{E}[g]$
- we use $\mathbb{E}[f] = \mathbb{E}[f - \lambda g] + \lambda\,\mathbb{E}[g]$ and choose λ to minimise the variance $\mathbb{V}[f - \lambda g]$ and hence the number of MC samples needed to estimate $\mathbb{E}[f - \lambda g]$

The difference with two-level MLMC is

- we use $\lambda = 1$ because $g \approx f$
- we use MC to estimate $\mathbb{E}[g]$, but it doesn't cost much

Two-level Monte Carlo

Three examples of using two-level Monte Carlo:

1) let $\widehat{P}_1$ be simulation using 32-bit floating point precision, and let $\widehat{P}_0$ be 16-bit calculation with same random numbers
– potentially factor $2\times$ performance benefit on latest CPUs

2) let $\widehat{P}_1$ be simulation using expensive random numbers from a "nasty" distribution (e.g. non-central chi-squared distribution for CIR interest rate and Heston stochastic volatility models), and $\widehat{P}_0$ be calculation using cheap random numbers from a similar distribution

   https://en.wikipedia.org/wiki/Noncentral_chi-squared_distribution

3) let $\widehat{P}_1$ be based on an expensive Navier-Stokes simulation, and $\widehat{P}_0$ on a much cheaper Euler simulation – this approach is usually referred to as Multi-Fidelity Monte Carlo

Multilevel Monte Carlo

Natural generalisation: given a sequence $\widehat{P}_0, \widehat{P}_1, \ldots, \widehat{P}_L$

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

we can use the estimator

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

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

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

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

gives

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

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 $\widehat{P}_\ell$ 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 MSE $= \displaystyle\sum_{\ell=0}^{L} N_\ell^{-1} V_\ell + \Big(\mathbb{E}[\widehat{P}_L] - \mathbb{E}[P] \Big)^2$

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 $\Big(\mathbb{E}[\widehat{P}_L] - \mathbb{E}[P] \Big)^2 < \frac{1}{2}\,\varepsilon^2$

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 Meta Theorem

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

If there exist independent estimators $\widehat{Y}_\ell$ 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

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

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

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

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

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

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

with an expected computational cost C with bound

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

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 $\ell \to \infty$. 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.

Financial application

- basket of 5 underlying assets, modelled by Geometric Brownian Motion

$$\mathrm{d}S_i = r\, S_i\, \mathrm{d}t + \sigma_i\, S_i\, \mathrm{d}W_i$$

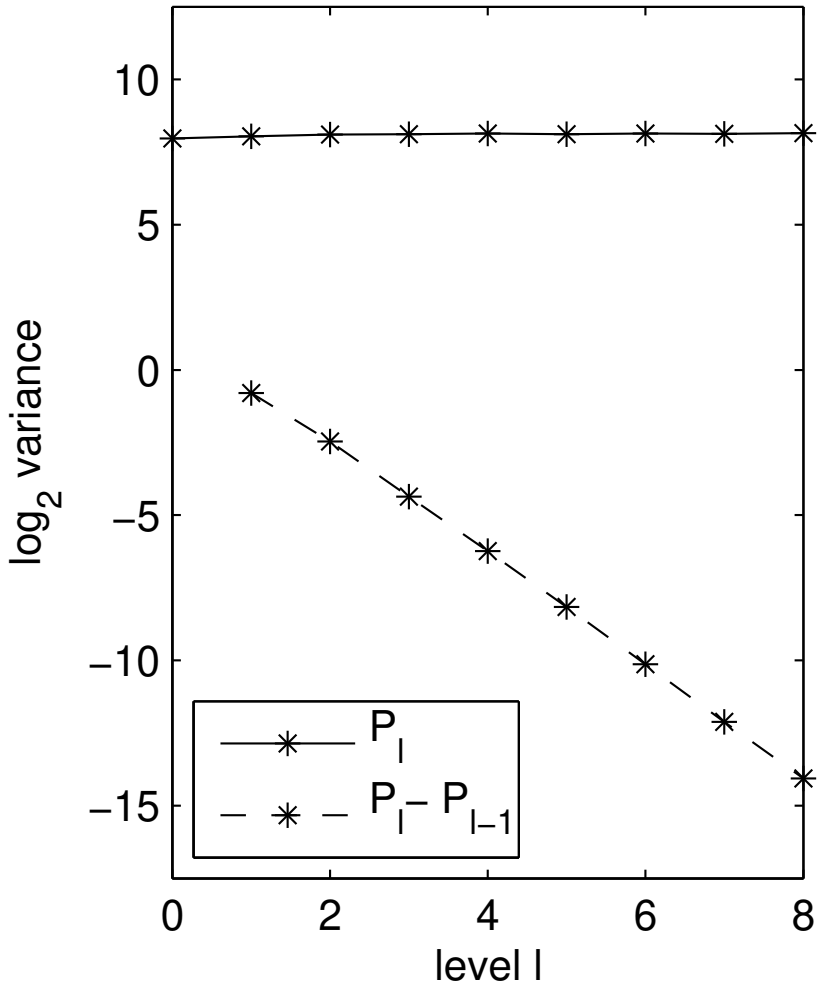
  with correlation between 5 driving Brownian motions

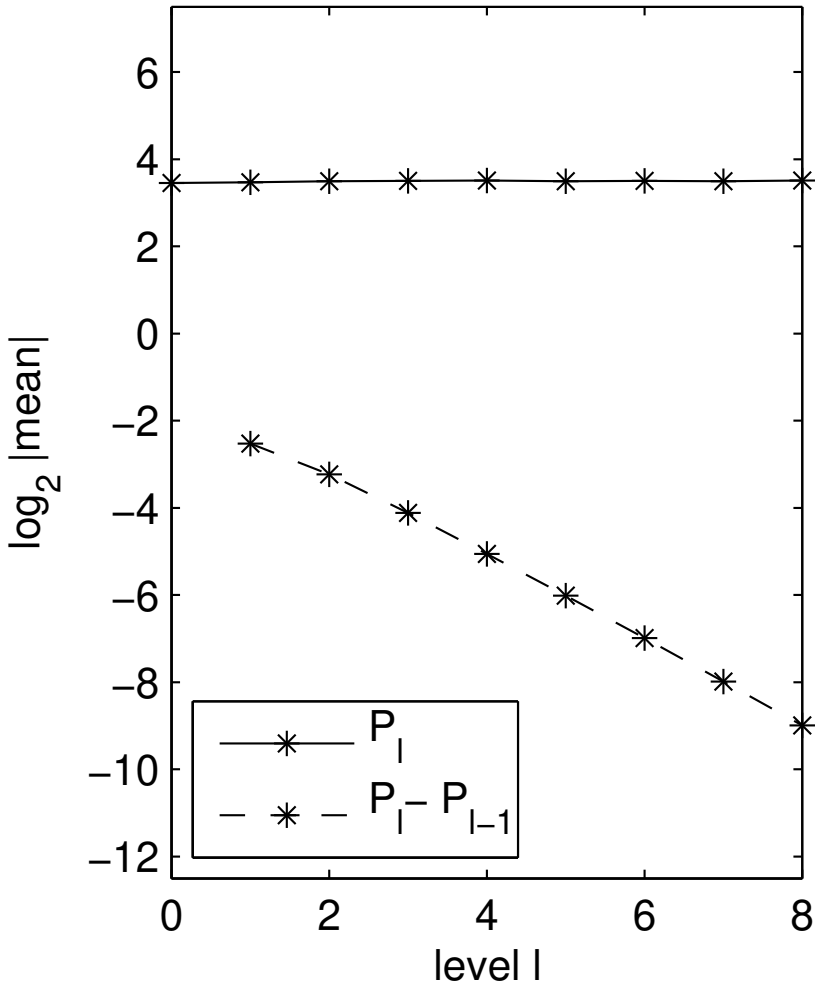
- Milstein numerical approximation

- call option is piecewise linear function of average at final time T

Financial application

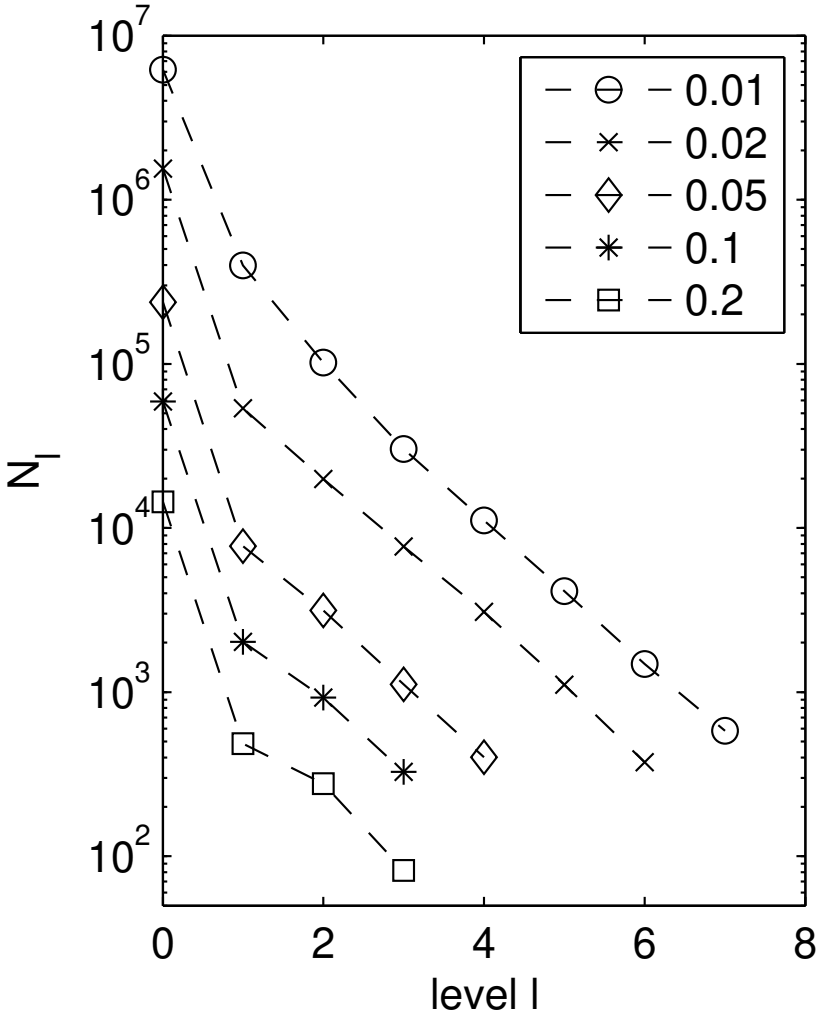
Basket call option:

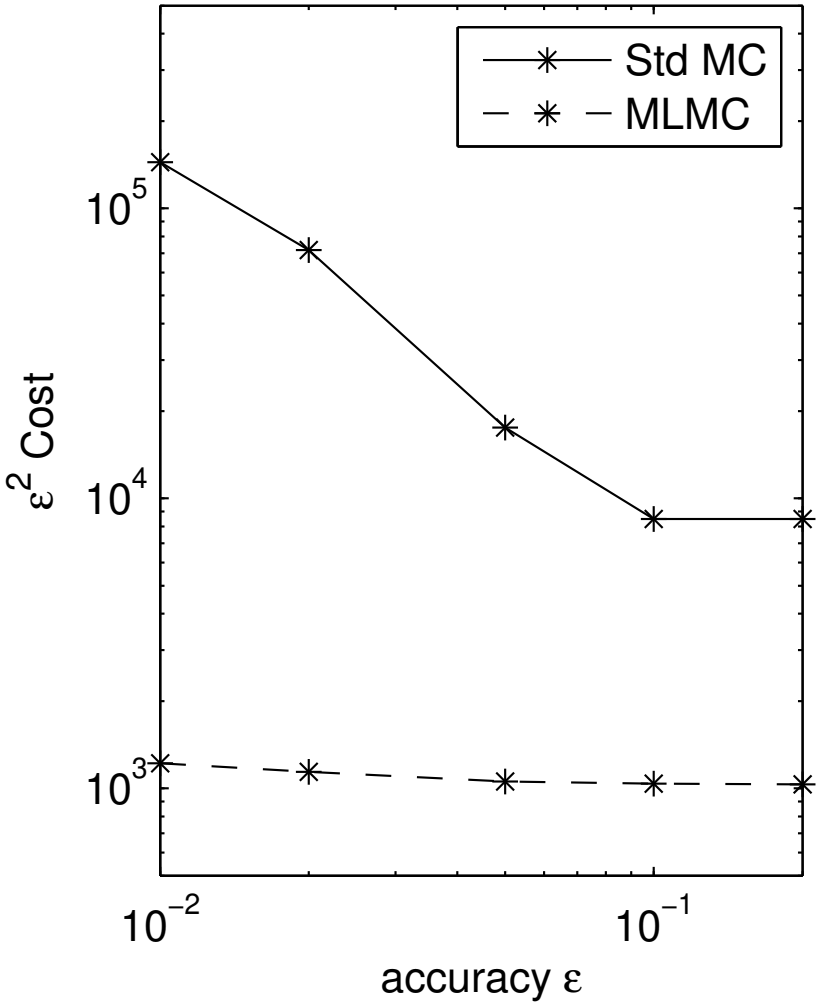




Financial application

Basket call option:





MLMC

Numerical algorithm:

1. start with $L=0$
2. if $L<2$, get an initial estimate for V_L using $N_L=1000$ samples, otherwise extrapolate from earlier levels
3. for $\ell \le L$, determine optimal N_ℓ to achieve $\sum_{\ell=0}^{L} V_\ell/N_\ell \le \varepsilon^2/2$
4. perform extra calculations as needed, updating estimates of V_ℓ
5. if $L<2$ or the bias estimate is greater than $\varepsilon/\sqrt{2}$, set $L := L+1$ and go back to step 2

MLMC generalisation

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

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

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

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 $f(\lambda) \equiv \mathbb{E}[g(x, \lambda)]$ 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, Farrell, Reisinger (Oxford) – parabolic, elliptic
  - Cliffe, G, Scheichl, Teckentrup (Bath/Nottingham) – elliptic
  - Jenny, Mishra, Schwab (ETH Zürich) – elliptic, parabolic, hyperbolic
  - Barth (Stuttgart) – elliptic
  - Lang (Chalmers) – parabolic
  - Dick, Kuo, Sloan (UNSW) – MLQMC for elliptic
  - Nuyens, Vandewalle (Leuven) – elliptic, engineering applications
  - Harbrecht, Peters (Basel) – elliptic
  - Haji-Ali, Nobile, Tempone (EPFL, KAUST) – elliptic
  - Chernov (Oldenberg) – elliptic
  - Ullmann (Munich) – elliptic
  - Efendiev (Texas A&M) – numerical homogenization
  - Heitzinger (Vienna) – PDEs in nanotechnology

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

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

- 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

PDEs with Uncertainty

I worked with Rob Scheichl (then Bath, now Heidelberg) and Andrew Cliffe (Nottingham) on multilevel Monte Carlo for the modelling of oil reservoirs and groundwater contamination in nuclear waste repositories.

Here we have an elliptic SPDE coming from Darcy's law:

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

where the permeability $\kappa(x)$ is uncertain, and $\log \kappa(x)$ is often modelled as being Normally distributed with a spatial covariance such as

$$\mathrm{cov}(\log \kappa(x_1), \log \kappa(x_2)) = \sigma^2 \exp(-\|x_1 - x_2\|/\lambda)$$

Ian Sloan, Frances Kuo and Josef Dick have subsequently worked with Christoph Schwab on Multilevel QMC for this application.

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

(Have already mentioned Multi-fidelity Monte Carlo which is one class of non-geometric MLMC, but usually with very few 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 $K \Longrightarrow$ 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 expectation

Another class of MLMC applications is for nested expectations of the form $\mathbb{E}\left[\, f\left(\mathbb{E}[Z|X]\right)\right]$ – this arises in risk estimation (finance) and decision-making under uncertainty (EVPPI/EVSI).

A standard Monte Carlo approach would use N outer samples of X, and, for each one, M inner samples of Z so the combined estimator is

$$\frac{1}{N}\sum_{n=1}^{N} f\left(\frac{1}{M}\sum_{m=1}^{M} Z_{m,n}\right)$$

To achieve ε RMS accuracy requires $N = O(\varepsilon^{-2})$ and $M = O(\varepsilon^{-1})$, so the total cost is $O(\varepsilon^{-3})$.

Nested expectation

A simple MLMC treatment uses $M_\ell = 2^\ell M_0$ inner samples on level ℓ, so the cost C_ℓ is $O(M_\ell)$ and the estimator for a given outer sample X is

$$\widehat{Y}_\ell = f\left(\frac{1}{M_\ell}\sum_{m=1}^{M_\ell} Z^{(m)}\right) - f\left(\frac{1}{M_{\ell-1}}\sum_{m=1}^{M_{\ell-1}} Z^{(m+M_\ell)}\right)$$

with the $Z^{(m)}$ all generated independently conditional on X.

If $\mathbb{V}[Z|X]$ is finite and uniformly bounded, and f is Lipschitz, then $\widehat{Y}_\ell = O(M_\ell^{-1/2})$ so $V_\ell = O(M_\ell^{-1})$ and the complexity is $O(\varepsilon^{-2}|\log\varepsilon|^2)$.

Nested expectation

An improved "antithetic" MLMC estimator uses

$$\widehat{Y}_\ell = f\left(\frac{1}{M_\ell}\sum_{m=1}^{M_\ell} Z^{(m)}\right) - \tfrac{1}{2} f\left(\frac{1}{M_{\ell-1}}\sum_{m=1}^{M_{\ell-1}} Z^{(m)}\right) - \tfrac{1}{2} f\left(\frac{1}{M_{\ell-1}}\sum_{m=1}^{M_{\ell-1}} Z^{(m+M_{\ell-1})}\right)$$

with the $Z^{(m)}$ generated independently conditional on X.

$\widehat{Y}_\ell = 0$ if f is linear, and more generally if f has a bounded second derivative then $V_\ell = O(M_\ell^{-2})$ and the complexity is $O(\varepsilon^{-2})$.

(This is a good example of the "tricks" which have been developed to improve the MLMC variance – other techniques are needed if f is discontinuous)

Other MLMC applications

- parametric integration, integral equations (Heinrich)
- multilevel QMC (Dick, G, Kuo, Scheichl, Schwab, Sloan)
- stochastic chemical reactions (Anderson & Higham, Tempone)
- mixed precision computation on FPGAs (Korn, Ritter, Wehn)
- MLMC for MCMC (Scheichl, Schwab, Stuart, Teckentrup)
- nested simulation (G, Goda, Haji-Ali/Tempone, Hambly/Reisinger)
- invariant distribution of contractive Markov process (Glynn & Rhee)
- invariant distribution of ergodic SDEs (Fang, G)
- McKean-Vlasov equations (Haji-Ali, Szpruch)
- stochastic approximation (Frikha, Dereich)
- machine learning (Gerstner)
- numerical linear algebra (Acebron, Wu, Frommer)

Conclusions

- multilevel idea is very simple; key questions are how to apply it in new situations, and how to perform 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 some 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

Webpages

- research papers and talks:
  `people.maths.ox.ac.uk/gilesm/mlmc.html`
  `people.maths.ox.ac.uk/gilesm/slides.html`
- 70-page 2015 *Acta Numerica* review and MATLAB test codes:
  `people.maths.ox.ac.uk/gilesm/acta/`
- MATLAB and C++ software for lots of applications:
  `people.maths.ox.ac.uk/gilesm/mlmc/`
- community webpage listing groups and research papers using MLMC:
  `people.maths.ox.ac.uk/gilesm/mlmc_community.html`