Multilevel Monte Carlo methods

Mike Giles

Mathematical Institute, University of Oxford

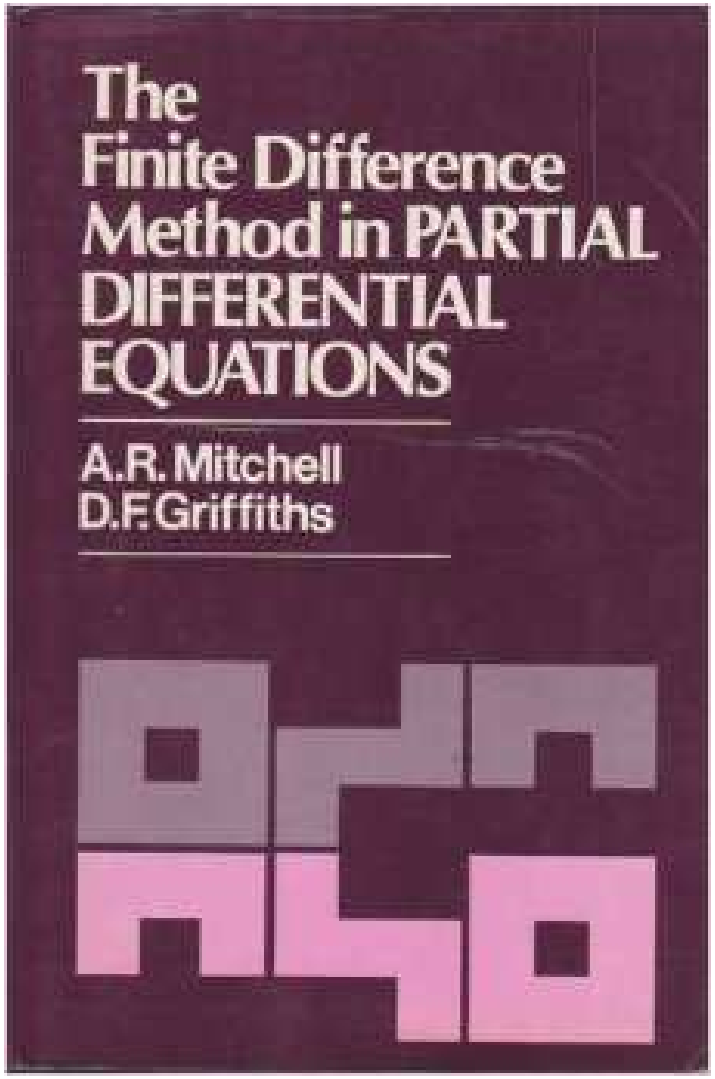
A R Mitchell Lecture

26th Biennial Numerical Analysis Conference, 2015

Thanks to many collaborators: Des Higham, Xuerong Mao (Strathclyde), Frances Kuo, Ian Sloan, Ben Waterhouse (UNSW), Kristian Debrabant, Andreas Rößler (Darmstadt), Rob Scheichl, Aretha Teckentrup, Elisabeth Ullmann (Bath), Andrew Cliffe (Nottingham), Klaus Ritter, Tigran Nagapetyan (Kaiserslautern), Sylvestre Burgos, Yuan Xia, Christoph Reisinger, Ruth Baker, Chris Lester, Kit Yates, James Whittle, Endre Süli, Shenghan Ye (Oxford), Lukas Szpruch (Edinburgh), Ferran Vidal, Jaime Peraire, Ngoc Cuong Nguyen (MIT), ...

An outline history

- inspired by undergraduate numerical projects course at Cambridge, and summer projects at Rolls-Royce
- this was one of my first textbooks
- after 25 years working on CFD, 10 years ago I switched to Monte Carlo methods for computational finance and other application areas



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
- the scope for improved performance through being creative
- 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

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

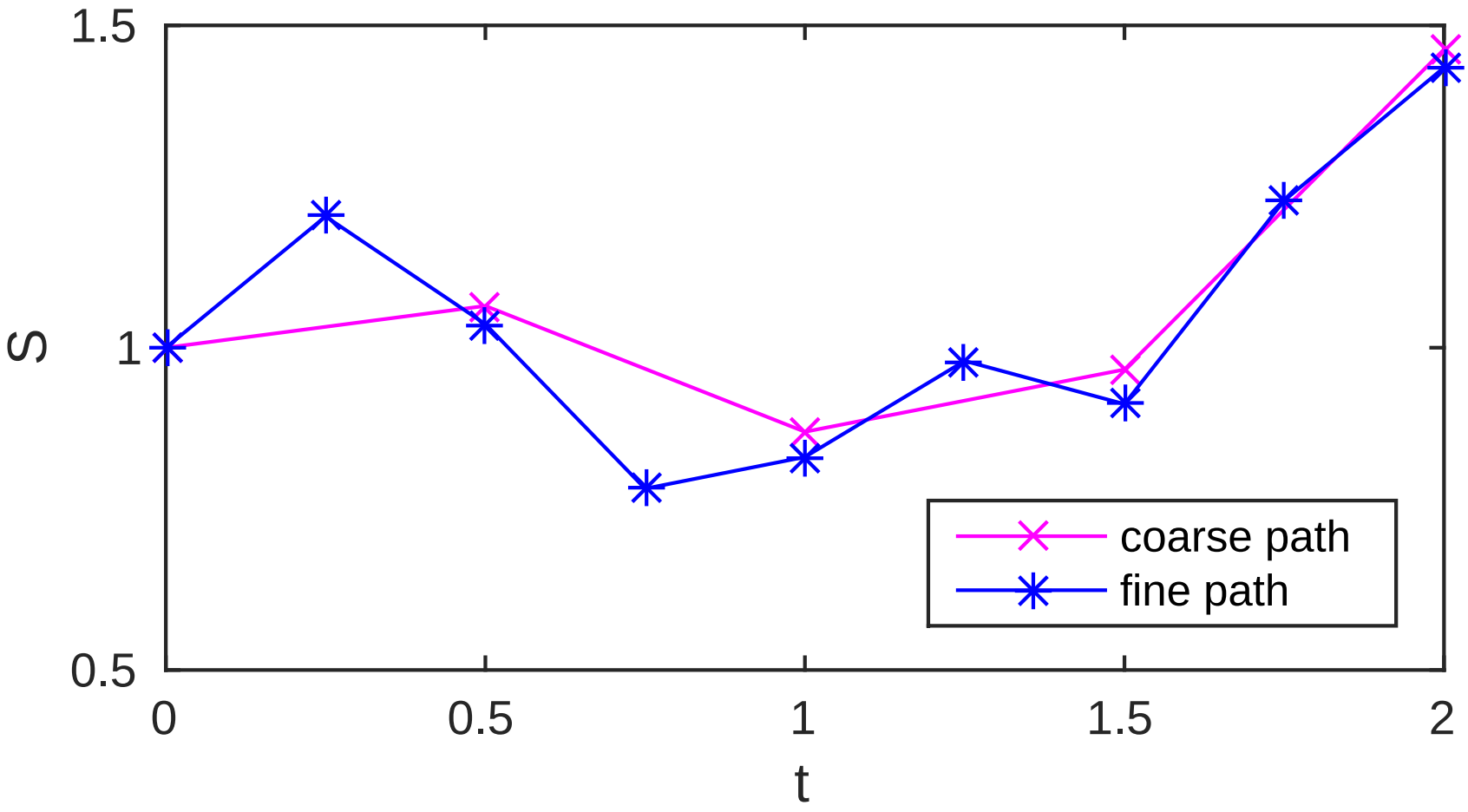
with uniform timestep h, and increments ΔW_n with variance h.

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.

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 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.

MLMC generalisation

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

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

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

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 Challenges

- not always obvious how to couple coarse and fine levels
  i.e. what does $\widehat{P}_\ell(\omega^{(n)}) - \widehat{P}_{\ell-1}(\omega^{(n)})$ mean?

- some creativity required to handle discontinuous functionals, where a small difference between the underlying coarse and fine simulations can produce an $O(1)$ difference in the output

- numerical analysis to determine the decay rate of V_ℓ can be tough

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

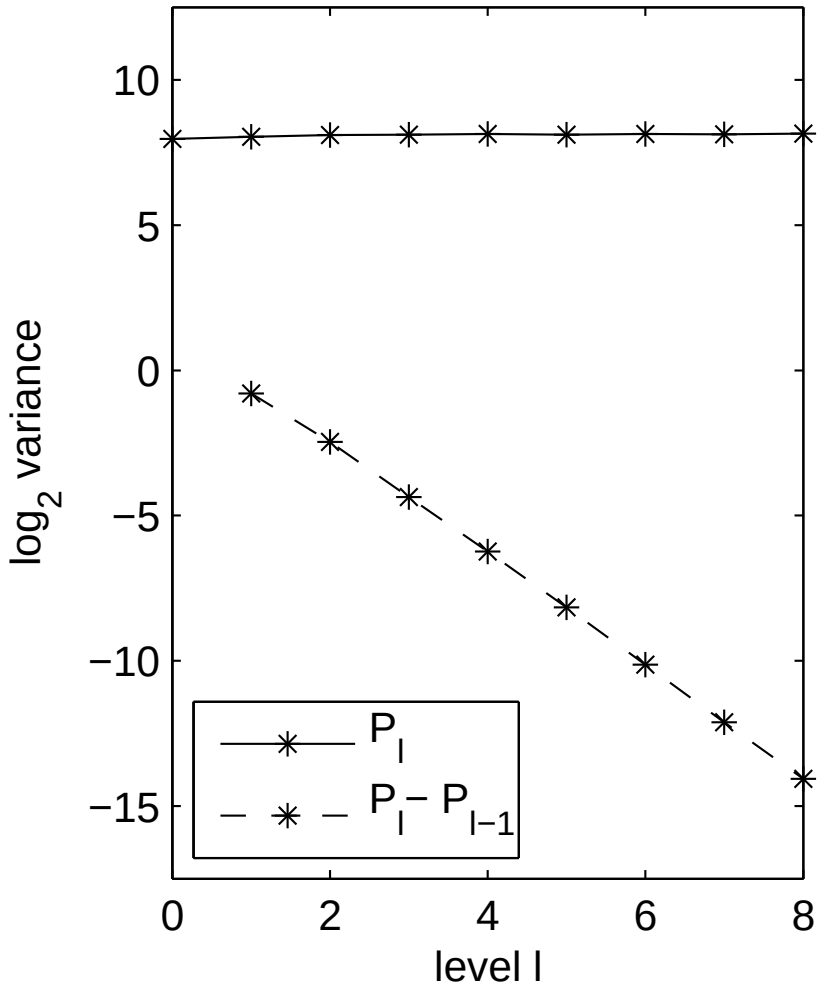
- Milstein numerical approximation

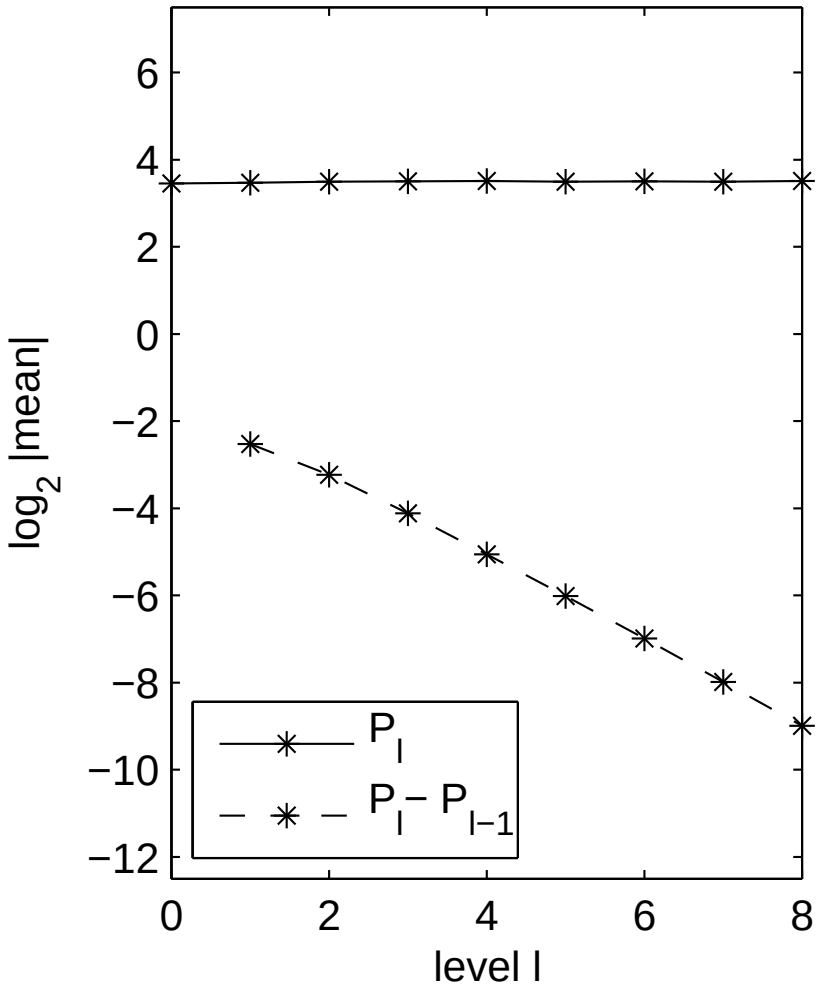
- standard call option is piecewise linear function of average at final time T

- digital call option is discontinuous function of average

Financial application

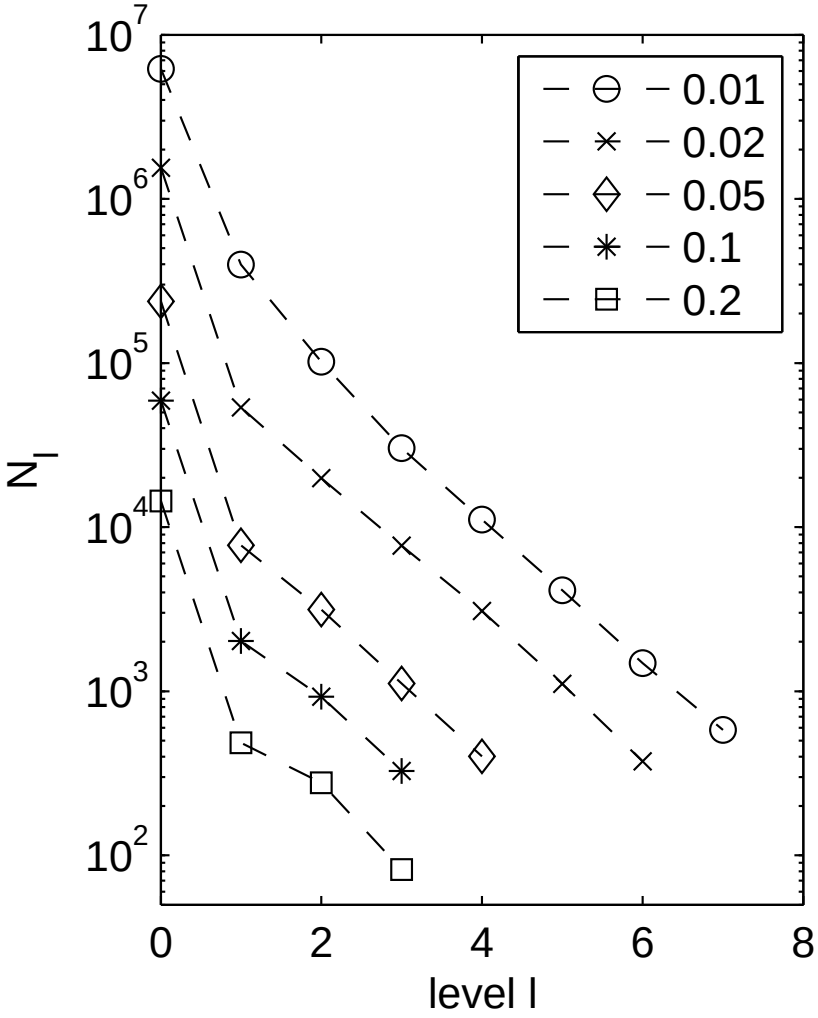
Standard call option:

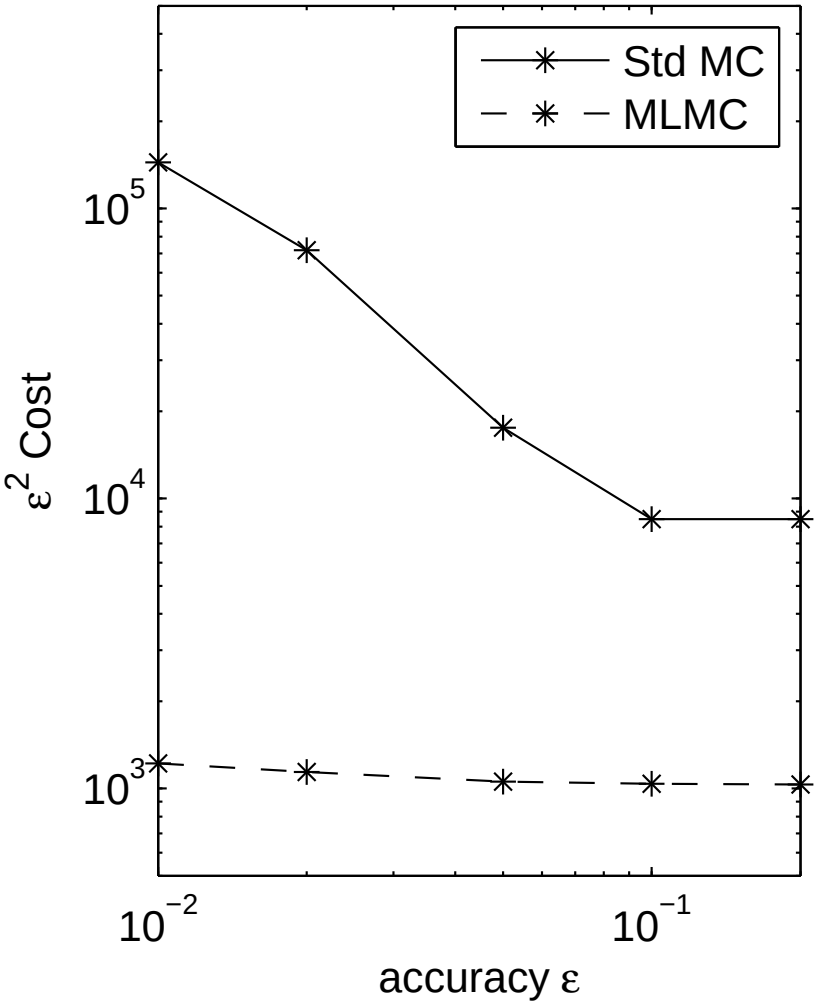




Financial application

Standard call option:





Digital options

In a digital option, the payoff is a discontinuous function of the final state.

Using the Milstein approximation, first order strong convergence means that $O(h_\ell)$ of the simulations have coarse and fine paths on opposite sides of a discontinuity.

Hence,

$$\widehat{P}_\ell - \widehat{P}_{\ell-1} = \begin{cases} O(1), & \text{with probability } O(h_\ell) \\ O(h_\ell), & \text{with probability } O(1) \end{cases}$$

so

$$\mathbb{E}[\widehat{P}_\ell - \widehat{P}_{\ell-1}] = O(h_\ell), \quad \mathbb{E}[(\widehat{P}_\ell - \widehat{P}_{\ell-1})^2] = O(h_\ell),$$

and hence $V_\ell = O(h_\ell)$, not $O(h_\ell^2)$

Digital options

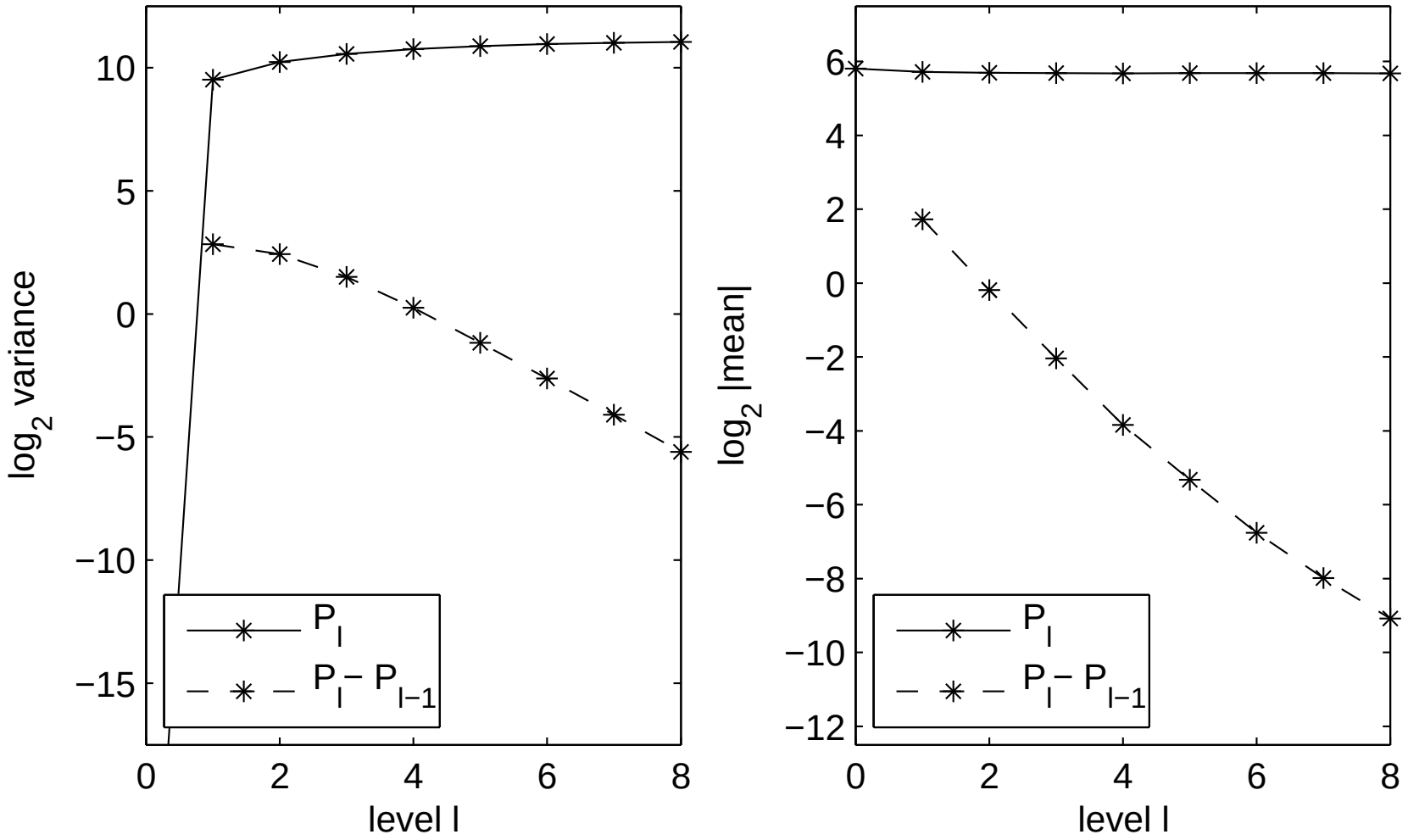
Three fixes:

- Conditional expectation: using the Euler discretisation instead of Milstein for the final timestep, conditional on all but the final Brownian increment, the final state has a Gaussian distribution, with a known analytic conditional expectation in simple cases
- Splitting: split each path simulation into M paths by trying M different values for the Brownian increment for the last fine path timestep
- Change of measure: when the expectation is not known, can use a change of measure so the coarse path takes the same final state as the fine path — difference in the "payoff" now comes from the Radon-Nikodym derivative

These all effectively smooth the payoff – end up with $V_\ell = O(h_\ell^{3/2})$.

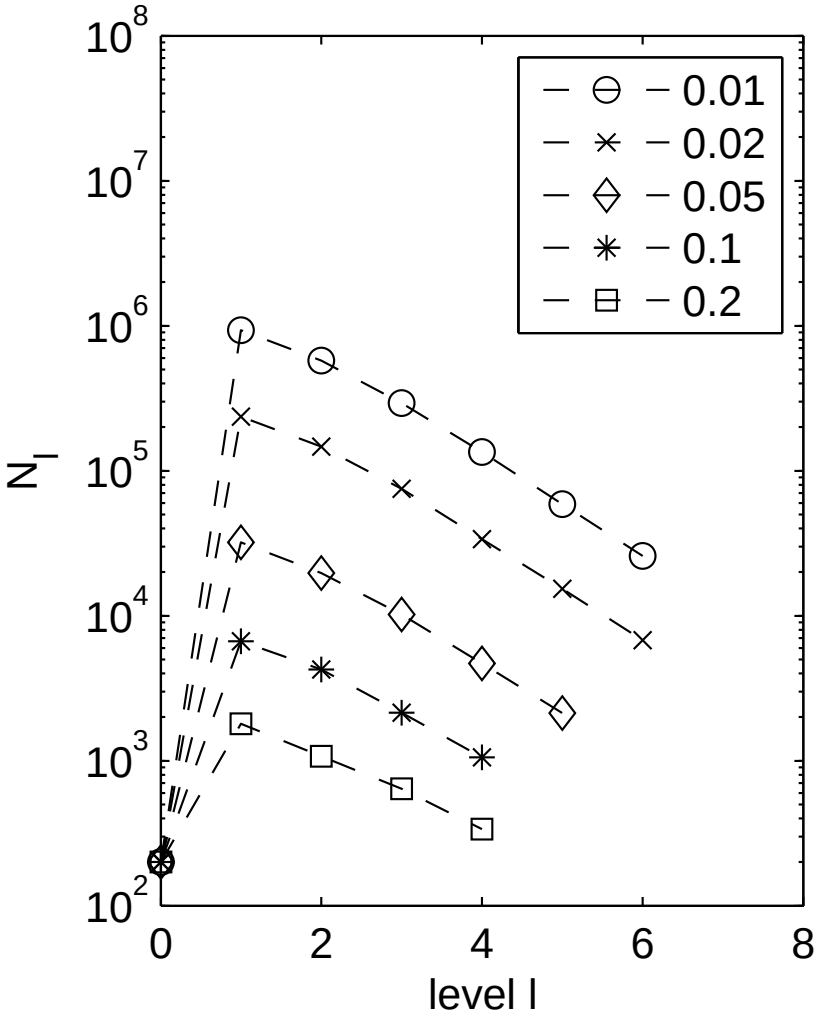
Financial application

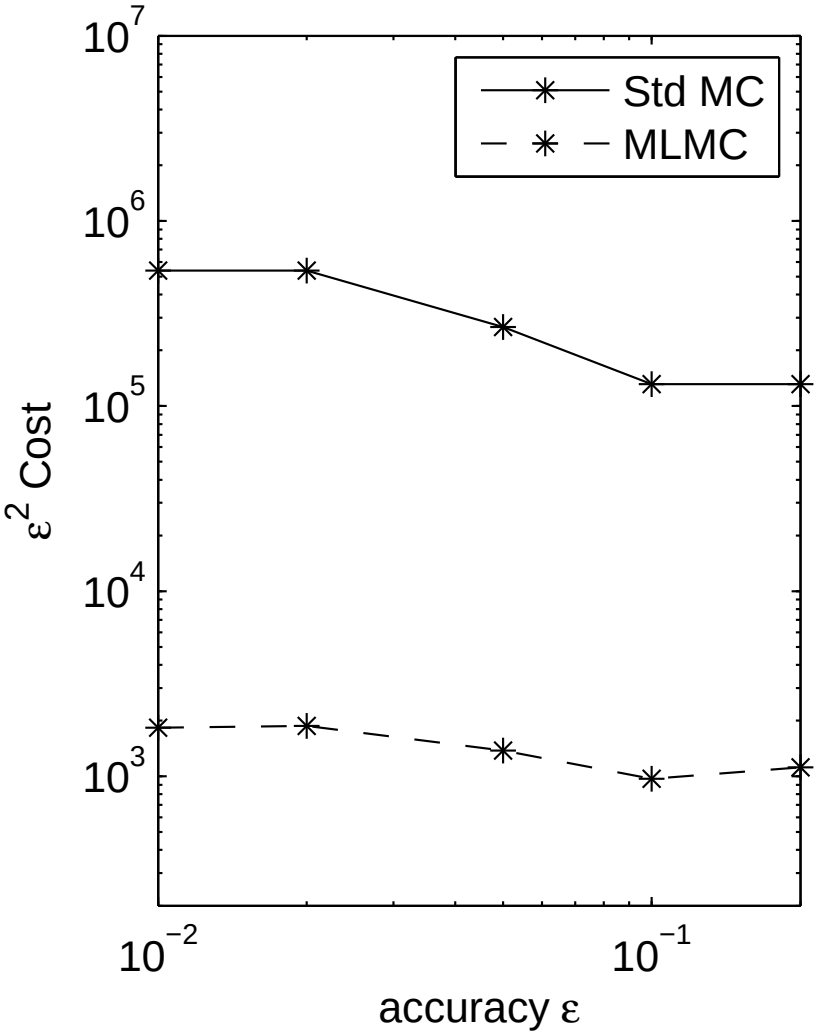
Digital call option:



Financial application

Digital call option:





Numerical Analysis

	Euler		Milstein	
option	numerics	analysis	numerics	analysis
Lipschitz	$O(h)$	$O(h)$	$O(h^2)$	$O(h^2)$
Asian	$O(h)$	$O(h)$	$O(h^2)$	$O(h^2)$
lookback	$O(h)$	$O(h)$	$O(h^2)$	$o(h^{2-\delta})$
barrier	$O(h^{1/2})$	$o(h^{1/2-\delta})$	$O(h^{3/2})$	$o(h^{3/2-\delta})$
digital	$O(h^{1/2})$	$O(h^{1/2}\lvert\log h\rvert)$	$O(h^{3/2})$	$o(h^{3/2-\delta})$

Table: V_ℓ convergence observed numerically (for GBM) and proved analytically (for more general SDEs)

Euler analysis due to G, Higham & Mao (2009) and Avikainen (2009).
Milstein analysis due to G, Debrabant & Rößler (2012).

Other MLMC work on SDEs

- 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
  - Cliffe, G, Scheichl, Teckentrup (Bath/Nottingham) – elliptic
  - Barth, Jenny, Lang, Meyer, Mishra, Müller, Schwab, Sukys, Zollinger (ETH Zürich) – elliptic, parabolic, hyperbolic
  - Harbrecht, Peters (Basel) – elliptic
  - Efendiev (Texas A&M) – numerical homogenization
  - Vidal-Codina, G, Peraire (MIT) – reduced basis approximation
  - G, Hou, Zhang (Caltech) – numerical homogenization

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

$$\widehat{P}_\ell(\omega) - P(\omega) \approx c(\omega)\, 2^{-2\ell}$$

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

- 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

Stochastic chemical reactions

In stochastic chemical reaction simulations, each reaction is a Poisson process with a rate which depends on the current concentrations.

In the "tau-leaping" method the reaction rates are frozen at the start of the timestep, so for each reaction need to sample from a Poisson process

$$P(\lambda\,\Delta t)$$

to determine the number of reactions in that timestep.

(As $\lambda\,\Delta t \to \infty$, the standard deviation becomes smaller relative to the mean, and it approaches the deterministic limit.)

Stochastic chemical reactions

Anderson & Higham (2011) have developed a very efficient multilevel version of this algorithm – big savings because finest level usually has 1000's of timesteps.

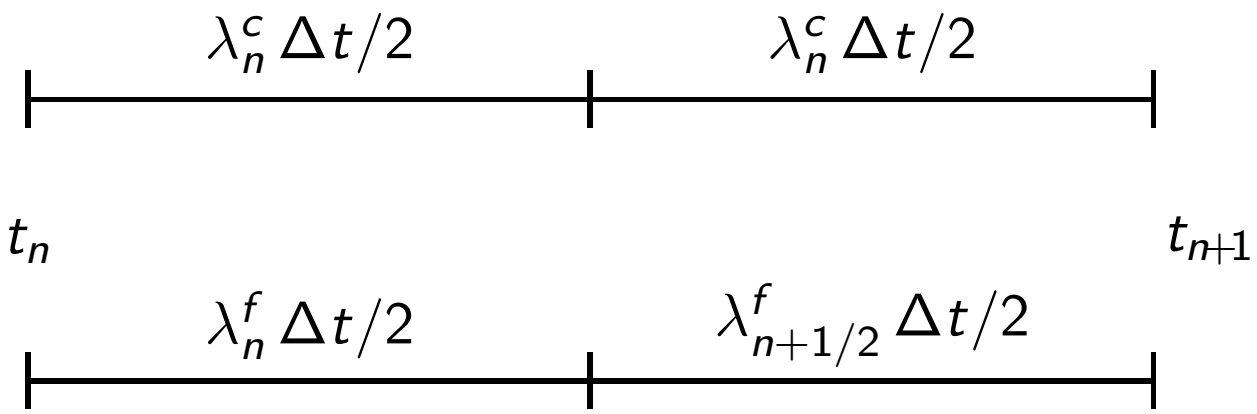
Key challenge: how to couple coarse and fine path simulations?

Crucial observation: $P(t_1) + P(t_2) \overset{d}{=} P(t_1 + t_2) \quad \text{for } t_1, t_2 \geq 0$

Stochastic chemical reactions

Solution:

- simulate the Poisson variable on the coarse timestep as the sum of two fine timestep Poisson variables
- couple the fine path and coarse path Poisson variables by using common variable based on smaller of two rates



If $\lambda_n^c > \lambda_n^f$, use $P(\lambda_n^c \Delta t/2) \sim P(\lambda_n^f \Delta t/2) + P((\lambda_n^c - \lambda_n^f)\Delta t/2)$

Other MLMC applications

- parametric integration, integral equations (Heinrich)
- multilevel QMC (Dick, G, Kuo, Scheichl, Schwab, Sloan)
- mixed precision computation on FPGAs (Korn, Ritter, Wehn)
- MLMC for MCMC (Scheichl, Schwab, Stuart, Teckentrup)
- Coulomb collisions in plasma (Caflisch)
- nested simulation (Haji-Ali & Tempone, Hambly & Reisinger)
- invariant distribution of contractive Markov process (Glynn & Rhee)
- invariant distribution of contractive SDEs (G, Lester & Whittle)

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 (2013)
  - 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

Feynman-Kac project with Francisco Bernal

Suppose that X_t satisfies the SDE

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

in bounded domain D, where W_t is an uncorrelated Brownian motion, and let

$$u(x,t) = \mathbb{E}\left[\int_t^\tau E(t,s)\,f(X_s,s)\,\mathrm{d}s \;+\; E(t,\tau)\,g(X_\tau,\tau) \mid X_t = x\right]$$

where τ is the first time at which X_t crosses the boundary ∂D, and

$$E(t_0,t_1) = \exp\left(-\int_{t_0}^{t_1} V(X_t,t)\,\mathrm{d}t\right).$$

Feynman-Kac formula

If $f(x,t), g(x,t), V(x,t), a(x,t), b(x,t)$ are all Lipschitz continuous, then the Feynman-Kac formula states that $u(x,t)$ satisfies the PDE

$$\frac{\partial u}{\partial t} + \sum_j a_j \frac{\partial u}{\partial x_j} + \tfrac{1}{2} \sum_{j,k,l} b_{j,l} b_{k,l} \frac{\partial^2 u}{\partial x_j \partial x_k} - V\, u + f = 0$$

in domain D, subject to $u(x,t) = g(x,t)$ on the boundary ∂D.

Hence, can estimate the solution to a high-dimensional PDE at a particular point (x,t) through Monte Carlo simulation of the SDE.

This also extends to linear and nonlinear functionals of the PDE solution.

Numerical approximation

Let $\widehat{X}_t$ be the piecewise-constant Euler-Maruyama approximation and define

$$\widehat{E}(t_0, t_1) = \exp\left(-\int_{t_0}^{t_1} V(\widehat{X}_t, t)\,\mathrm{d}t\right),$$

and let

$$\widehat{u}(x,t) = \mathbb{E}\left[\int_t^{\widehat{\tau}} \widehat{E}(t,s)\, f(\widehat{X}_s, s)\,\mathrm{d}s + \widehat{E}(t,\widehat{\tau})\, g(\widehat{X}_{\widehat{\tau}}, \widehat{\tau}) \mid \widehat{X}_t = x\right],$$

with the Euler-Maruyama discretisation beginning at time t, and with $\widehat{\tau}$ being the exit time.

Numerical approximation

The Euler-Maruyama method has strong accuracy $O(h^{1/2})$, and the natural definition of τ gives an $O(h^{1/2})$ weak error too.

For standard Monte Carlo method, ε RMS accuracy needs $O(\varepsilon^{-2})$ paths, each with $h = O(\varepsilon^2)$, so total cost is $O(\varepsilon^{-4})$

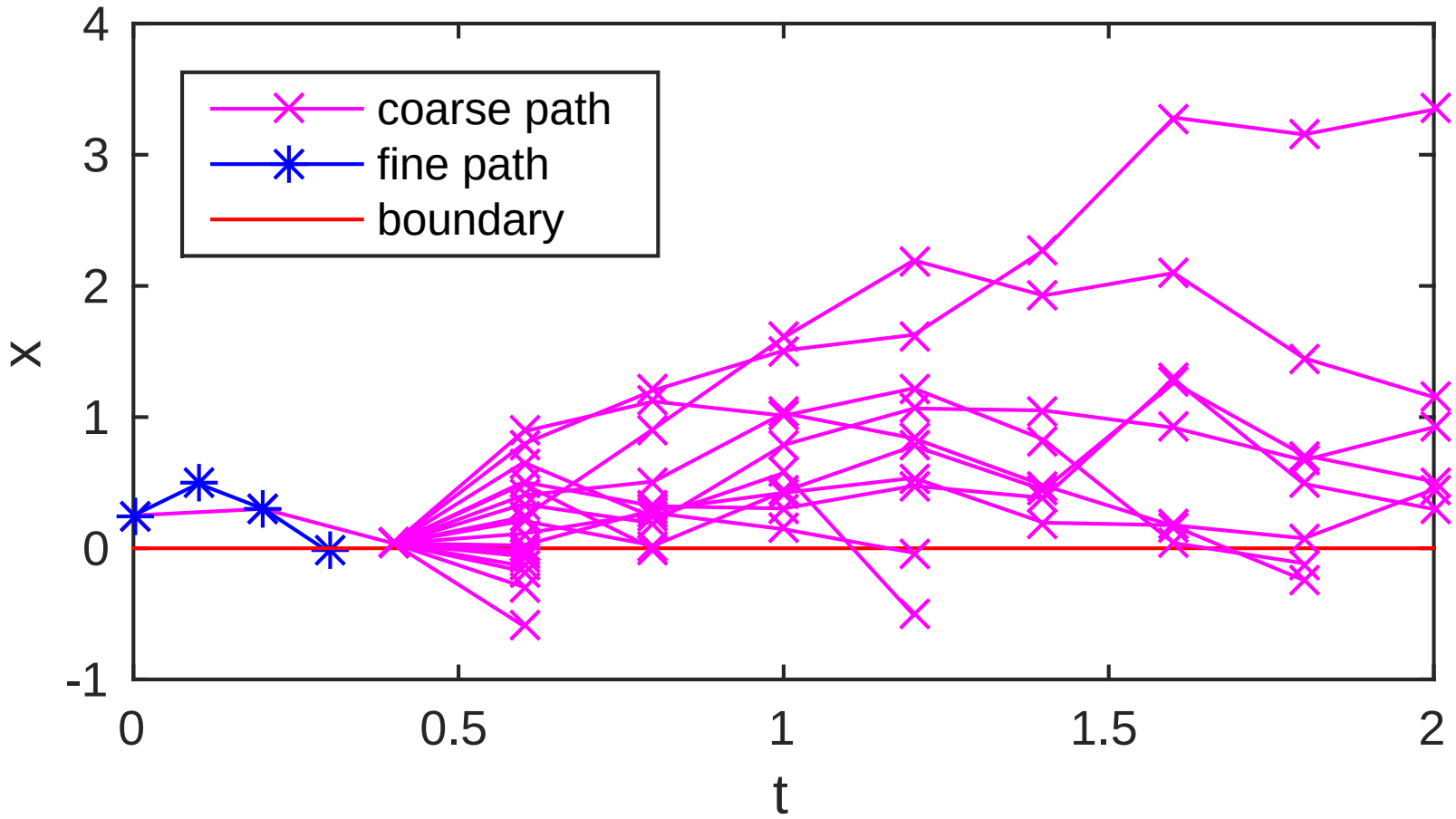
Gobet reduced this to $O(\varepsilon^{-3})$ by shifting the boundary by $O(h^{1/2})$ to improve the weak accuracy to $O(h)$.

Alternatively, Higham *et al* use MLMC to achieve $O(\varepsilon^{-3}|\log\varepsilon|^3)$ complexity without shifting the boundary.

MLMC challenge

When coarse or fine path exits the domain, the other is within $O(h^{1/2})$.

However, there is a $O(h^{1/2})$ probability that it will not exit the domain until much later $\Longrightarrow V_\ell = O(h^{1/2})$.



MLMC challenge

How can we do better?

Similar to previous work on digital options, split second path into multiple copies, and average their outputs to approximate the conditional expectation – see also recent work by Dickmann & Schweizer (2014).

$O(h^{1/2})$ expected time to exit for second path, so can afford to use $O(h^{-1/2})$ copies of second path.

This gives an approximation to the conditional expectation resulting in $\widehat{P}_\ell - \widehat{P}_{\ell-1} \approx O(h^{1/2})$, so $V_\ell \approx O(h_\ell)$.

Numerical results confirm this, and supporting numerical analysis is almost complete.

Conclusions

- multilevel idea is very simple; key question is how to apply it in new situations, and 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 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 new 70-page *Acta Numerica* review and MATLAB test codes:

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

– contains references to almost all MLMC research, including some very early related work by Achi Brandt

MLMC Community

Webpage: people.maths.ox.ac.uk/gilesm/mlmc_community.html

Abo Academi (Avikainen) – numerical analysis
Basel (Harbrecht) – elliptic SPDEs, sparse grids
Bath (Kyprianou, Scheichl, Shardlow, Yates) – elliptic SPDEs, MCMC, Lévy-driven SDEs, stochastic chemical modelling
Chalmers (Lang) – SPDEs
Duisburg (Belomestny) – Bermudan and American options
Edinburgh (Davie, Szpruch) – SDEs, numerical analysis
EPFL (Abdulle) – stiff SDEs and SPDEs
ETH Zürich (Jenny, Jentzen, Schwab) – SPDEs, multilevel QMC
Frankfurt (Gerstner, Kloeden) – numerical analysis, fractional Brownian motion
Fraunhofer ITWM (Iliev) – SPDEs in engineering
Hong Kong (Chen) – Brownian meanders, nested simulation in finance
IIT Chicago (Hickernell) – SDEs, infinite-dimensional integration, complexity analysis
Kaiserslautern (Heinrich, Korn, Ritter) – finance, SDEs, parametric integration, complexity analysis
KAUST (Tempone, von Schwerin) – adaptive time-stepping, stochastic chemical modelling
Kiel (Gnewuch) – randomized multilevel QMC
LPMA (Frikha, Lemaire, Pagès) – numerical analysis, multilevel extrapolation, finance applications
Mannheim (Neuenkirch) – numerical analysis, fractional Brownian motion
MIT (Peraire) – uncertainty quantification, SPDEs
Munich (Hutzenthaler) – numerical analysis
Oxford (Baker, Giles, Hambly, Reisinger) – SDEs, SPDEs, numerical analysis, finance applications, stochastic chemical modelling
Passau (Müller-Gronbach) – infinite-dimensional integration, complexity analysis
Stanford (Glynn) – numerical analysis, randomized multilevel
Strathclyde (Higham, Mao) – numerical analysis, exit times, stochastic chemical modelling
Stuttgart (Barth) – SPDEs
Texas A&M (Efendiev) – SPDEs in engineering
UCLA (Caflisch) – Coulomb collisions in physics
UNSW (Dick, Kuo, Sloan) – multilevel QMC
UTS (Baldeaux) – multilevel QMC
Warwick (Stuart, Teckentrup) – MCMC for SPDEs
WIAS (Friz, Schoenmakers) – rough paths, fractional Brownian motion, Bermudan options
Wisconsin (Anderson) – numerical analysis, stochastic chemical modelling