Acta Numerica (2015), pp. 259–328
doi:10.1017/S096249291500001X

© Cambridge University Press, 2015
Printed in the United Kingdom

Multilevel Monte Carlo methods

Michael B. Giles
Mathematical Institute, University of Oxford,
Woodstock Road, Oxford OX2 6GG, UK
E-mail: mike.giles@maths.ox.ac.uk

Monte Carlo methods are a very general and useful approach for the estimation of expectations arising from stochastic simulation. However, they can be computationally expensive, particularly when the cost of generating individual stochastic samples is very high, as in the case of stochastic PDEs. Multilevel Monte Carlo is a recently developed approach which greatly reduces the computational cost by performing most simulations with low accuracy at a correspondingly low cost, with relatively few simulations being performed at high accuracy and a high cost.

In this article, we review the ideas behind the multilevel Monte Carlo method, and various recent generalizations and extensions, and discuss a number of applications which illustrate the flexibility and generality of the approach and the challenges in developing more efficient implementations with a faster rate of convergence of the multilevel correction variance.

CONTENTS

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

1. Introduction

1.1. Stochastic modelling and Monte Carlo simulation

Stochastic modelling and simulation is a growing area in applied mathematics and scientific computing. One large application area is in computational finance, in the pricing of financial derivatives and quantitative risk management (Glasserman 2004, Asmussen and Glynn 2007). Another is uncertainty quantification in engineering and science, which has led to a new SIAM journal and associated annual conferences. Stochastic modelling is also important in diverse areas such as biochemical reactions (Anderson and Higham 2012) and plasma physics (Rosin *et al.* 2014).

When the dimensionality of the uncertainty (or the number of uncertain input parameters) is low, it can be appropriate to model the uncertainty using the Fokker–Planck PDE and use stochastic Galerkin, stochastic collocation or polynomial chaos methods (Xiu and Karniadakis 2002, Babuška, Tempone and Zouraris 2004, Babuška, Nobile and Tempone 2010, Gunzburger, Webster and Zhang 2014). When the level of uncertainty is low, and its effect is largely linear, then moment methods can be an efficient and accurate way in which to quantify the effects on uncertainty (Putko, Taylor, Newman and Green 2002). However, when the uncertainty is high-dimensional and strongly nonlinear, Monte Carlo simulation remains the preferred approach.

At its simplest, Monte Carlo simulation is extremely simple. To estimate $\mathbb{E}[P]$, a simple Monte Carlo estimate is just an average of values $P(\omega)$ for N independent samples ω coming from a given probability space $(\Omega, \mathcal{F}, \mathbb{P})$,

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

The variance of this estimate is $N^{-1}\,\mathbb{V}[P]$, so the root mean square (RMS) error is $O(N^{-1/2})$ and an accuracy of ε requires $N = O(\varepsilon^{-2})$ samples. This is the weakness of Monte Carlo simulation: its computational cost can be very high, particularly when each sample $P(\omega)$ might require the approximate solution of a PDE, or a computation with many time-steps.

One approach to addressing this high cost is the use of quasi-Monte Carlo (QMC) methods, in which the samples are not chosen randomly and independently, but are instead selected very carefully to reduce the error. In the best cases, the error may be $O(N^{-1})$, up to logarithmic terms, giving a very substantial reduction in the number of samples required for a given accuracy. The QMC approach was surveyed in *Acta Numerica 2013* (Dick, Kuo and Sloan 2013). In this article, we cover a different approach to improving the computational efficiency, the multilevel Monte Carlo (MLMC) method, and we include a brief discussion of the combination of the two approaches.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

1.2. Control variates and two-level MLMC

One of the classic approaches to Monte Carlo variance reduction is through the use of a control variate (Glasserman 2004). Suppose we wish to estimate $\mathbb{E}[f]$, and there is a control variate g which is well correlated to f and has a known expectation $\mathbb{E}[g]$. In that case, we can use the following unbiased estimator for $\mathbb{E}[f]$ based on N independent samples $\omega^{(n)}$:

$$N^{-1}\sum_{n=1}^{N}\bigl\{f(\omega^{(n)})-\lambda\bigl(g(\omega^{(n)})-\mathbb{E}[g]\bigr)\bigr\}.$$

The optimal value for λ is $\rho\sqrt{\mathbb{V}[f]\,/\,\mathbb{V}[g]}$, where ρ is the correlation between f and g, and the variance of the control variate estimator is reduced by factor $1-\rho^2$ compared to the standard estimator.

A two-level version of MLMC is very similar. If we want to estimate $\mathbb{E}[P_1]$ but it is much cheaper to simulate P_0 which approximates P_1, then since

$$\mathbb{E}[P_1]=\mathbb{E}[P_0]+\mathbb{E}[P_1-P_0]$$

we can use the unbiased two-level estimator

$$N_0^{-1}\sum_{n=1}^{N_0}P_0^{(n)}+N_1^{-1}\sum_{n=1}^{N_1}\bigl(P_1^{(n)}-P_0^{(n)}\bigr).$$

Here $P_1^{(n)}-P_0^{(n)}$ represents the difference between P_1 and P_0 for the same underlying stochastic sample $\omega^{(n)}$, so that $P_1^{(n)}-P_0^{(n)}$ is small and has a small variance; the precise construction depends on the application, and various examples will be shown later. The two key differences from the control variate approach are that the value of $\mathbb{E}[P_0]$ is not known, so has to be estimated, and we use $\lambda=1$.

If we define C_0 and C_1 to be the cost of computing a single sample of P_0 and P_1-P_0, respectively, then the total cost is $N_0\,C_0+N_1\,C_1$, and if V_0 and V_1 are the variance of P_0 and P_1-P_0, then the overall variance is $N_0^{-1}V_0+N_1^{-1}V_1$, assuming that $\sum_{n=1}^{N_0}P_0^{(n)}$ and $\sum_{n=1}^{N_1}\bigl(P_1^{(n)}-P_0^{(n)}\bigr)$ use independent samples.

Hence, treating the integers N_0,N_1 as real variables and performing a constrained minimization using a Lagrange multiplier, the variance is minimized for a fixed cost by choosing $N_1/\,N_0=\sqrt{V_1/C_1}\;/\sqrt{V_0/C_0}$.

1.3. Multilevel Monte Carlo

The multilevel generalization is quite natural: given a sequence $P_0,\dots,P_{L-1}$ which approximates P_L with increasing accuracy, but also increasing cost,

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

we have the simple identity

$$\mathbb{E}[P_L] = \mathbb{E}[P_0] + \sum_{\ell=1}^{L} \mathbb{E}[P_\ell - P_{\ell-1}],$$

and therefore we can use the following unbiased estimator for $\mathbb{E}[P_L]$:

$$N_0^{-1} \sum_{n=1}^{N_0} P_0^{(0,n)} + \sum_{\ell=1}^{L} \left\{ N_\ell^{-1} \sum_{n=1}^{N_\ell} \bigl(P_\ell^{(\ell,n)} - P_{\ell-1}^{(\ell,n)}\bigr) \right\}$$

where the inclusion of the level ℓ in the superscript (ℓ, n) indicates that independent samples are used at each level of correction.

If we define C_0, V_0 to be the cost and variance of one sample of P_0, and C_ℓ, V_ℓ to be the cost and variance of one sample of $P_\ell - P_{\ell-1}$, then the overall cost and variance of the multilevel estimator are $\sum_{\ell=0}^{L} N_\ell\, C_\ell$ and $\sum_{\ell=0}^{L} N_\ell^{-1}\, V_\ell$, respectively.

For a fixed cost, the variance is minimized by choosing N_ℓ to minimize

$$\sum_{\ell=0}^{L} \bigl(N_\ell^{-1}\, V_\ell + \mu^2 N_\ell\, C_\ell\bigr)$$

for some value of the Lagrange multiplier μ^2. This gives $N_\ell = \mu \sqrt{V_\ell \,/\, C_\ell}$. To achieve an overall variance of ε^2 then requires that

$$\mu = \varepsilon^{-2} \sum_{\ell=0}^{L} \sqrt{V_\ell\, C_\ell},$$

and the total computational cost is therefore

$$C = \varepsilon^{-2} \left(\sum_{\ell=0}^{L} \sqrt{V_\ell\, C_\ell} \right)^2. \tag{1.1}$$

It is important to note whether the product $V_\ell\, C_\ell$ increases or decreases with ℓ, that is, whether or not the cost increases with level faster than the variance decreases. If the product increases with level, so that the dominant contribution to the cost comes from $V_L\, C_L$, then we have $C \approx \varepsilon^{-2} V_L\, C_L$, whereas if it decreases and the dominant contribution comes from $V_0\, C_0$, then $C \approx \varepsilon^{-2} V_0\, C_0$. This contrasts with the standard MC cost of approximately $\varepsilon^{-2} V_0\, C_L$, assuming that the cost of computing P_L is similar to the cost of computing $P_L - P_{L-1}$, and that $\mathbb{V}[P_L] \approx \mathbb{V}[P_0]$.

This shows that in the first case the MLMC cost is reduced by factor V_L/V_0, corresponding to the ratio of the variances $\mathbb{V}[P_L - P_{L-1}]$ and $\mathbb{V}[P_L]$, whereas in the second case it is reduced by factor C_0/C_L, the ratio of the costs of computing P_0 and $P_L - P_{L-1}$. If the product $V_\ell\, C_\ell$ does not vary with level, then the total cost is $\varepsilon^{-2} L^2\, V_0\, C_0 = \varepsilon^{-2} L^2\, V_L\, C_L$.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

1.4. *Origins of MLMC method*

The first work on multilevel Monte Carlo methods was by Heinrich for parametric integration, the evaluation of functionals arising from the solution of integral equations, and weakly singular integral operators (Heinrich 1998, Heinrich and Sindambiwe 1999, Heinrich 2000, Heinrich 2001, Heinrich 2006). Parametric integration concerns the estimation of $\mathbb{E}[f(x,\lambda)]$, where x is a finite-dimensional random variable and λ is a parameter. In the simplest case in which λ is a scalar real variable in the range $[0,1]$, having estimated the value of $\mathbb{E}[f(x,0)]$ and $\mathbb{E}[f(x,1)]$, one can use $(f(x,0)+f(x,1))/2$ as a control variate when estimating the value of $\mathbb{E}[f(x,1/2)]$. Heinrich applied this idea recursively on a geometric sequence of levels, leading to a complexity analysis which is very similar to that which will be outlined later.

Although not so clearly related, there are also papers by Brandt *et al.* (Brandt, Galun and Ron 1994, Brandt and Ilyin 2003) which combine Monte Carlo techniques with multigrid ideas in determining thermodynamic limits in statistical physics applications.

Kebaier (2005) developed a two-level approach for path simulation which is very similar to the author's approach (Giles 2008*a*, 2008*b*), which was inspired by the multigrid ideas of Brandt and others for the iterative solution of PDE approximations. The differences are that Kebaier used only two levels, and used a general multiplicative factor as in the standard control variate approach. There was also similar independent work at the same time by Li (2007) and Speight (2009).

1.5. *Outline of article*

Section 2 presents the main theoretical results which underpin MLMC methods, and a number of different generalizations and extensions, many of which have been developed very recently.

Section 3 presents in some detail a MATLAB implementation of MLMC. This is used to generate the numerical results which are presented later, and the full MATLAB code is available online (Giles 2014).

Sections 4–10 cover a wide range of applications, demonstrating the flexibility and generality of the MLMC approach, documenting progress in the numerical analysis of the methods, and illustrating techniques which can be used to improve the variance of the MLMC estimators. Particular attention is paid to the difficulties caused by discontinuities in output functionals, and a variety of remedies which can be employed.

Finally, Section 11 concludes with some thoughts about directions for future research.

Readers who are new to the subject may wish to start with Section 2.1 and then proceed to the application sections most relevant to their interests.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

2. MLMC theory and extensions

2.1. Geometric MLMC

In the Introduction, we considered the case of a general multilevel method in which the output P_L on the finest level corresponds to the quantity of interest. However, in many infinite-dimensional applications, such as those which involve the simulation of SDEs and SPDEs, the output P_ℓ on level ℓ is an approximation to a random variable P which cannot be simulated exactly. If Y is an approximation to $\mathbb{E}[P]$, then a standard piece of theory gives the mean square error (MSE) as

$$\mathrm{MSE} \equiv \mathbb{E}\big[(Y-\mathbb{E}[P])^2\big] = \mathbb{V}[Y] + (\mathbb{E}[Y]-\mathbb{E}[P])^2. \tag{2.1}$$

If Y is now the multilevel estimator

$$Y = \sum_{\ell=0}^{L} Y_\ell, \quad Y_\ell = N_\ell^{-1} \sum_{n=1}^{N_\ell} \big(P_\ell^{(\ell,n)} - P_{\ell-1}^{(\ell,n)}\big), \tag{2.2}$$

with $P_{-1} \equiv 0$, then

$$\mathbb{E}[Y] = \mathbb{E}[P_L], \quad \mathbb{V}[Y] = \sum_{\ell=0}^{L} N_\ell^{-1} V_\ell, \quad V_\ell \equiv \mathbb{V}[P_\ell - P_{\ell-1}]. \tag{2.3}$$

To ensure that the MSE is less than ε^2, it is sufficient to ensure that $\mathbb{V}[Y]$ and $(\mathbb{E}[P_L - P])^2$ are both less than $\varepsilon^2/2$. Combining this idea with a geometric sequence of levels in which the cost increases exponentially with level, while both the weak error $\mathbb{E}[P_L - P]$ and the multilevel correction variance V_ℓ decrease exponentially, leads to the following theorem.

Theorem 2.1. Let P denote a random variable, and let P_ℓ denote the corresponding level ℓ numerical approximation.

If there exist independent estimators Y_ℓ based on N_ℓ Monte Carlo samples, each with expected cost C_ℓ and variance V_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) $|\mathbb{E}[P_\ell - P]| \leq c_1\, 2^{-\alpha\,\ell}$,

(ii) $\mathbb{E}[Y_\ell] = \begin{cases} \mathbb{E}[P_0] & \ell = 0, \\ \mathbb{E}[P_\ell - P_{\ell-1}] & \ell > 0, \end{cases}$

(iii) $V_\ell \leq c_2\, 2^{-\beta\,\ell}$,

(iv) $C_\ell \leq c_3\, 2^{\gamma\,\ell}$,

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

then there exists a positive constant c_4 such that for any $\varepsilon < e^{-1}$ there are values L and N_ℓ for which the multilevel estimator

$$Y = \sum_{\ell=0}^{L} Y_\ell$$

has a mean square error with bound

$$\mathrm{MSE} \equiv \mathbb{E}\big[(Y - \mathbb{E}[P])^2\big] < \varepsilon^2$$

with a computational complexity C with bound

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

The statement of the theorem is a slight generalization of the original theorem in Giles (2008*b*). It corresponds to the theorem and proof in Cliffe, Giles, Scheichl and Teckentrup (2011), except for the minor change to expected costs to allow for applications in which the simulation cost of individual samples is itself random. Note that if condition (iii) is tightened slightly to be a bound on $\mathbb{E}[(P_\ell - P_{\ell-1})^2]$, which is usually the quantity which is bounded in numerical analysis, then it would follow immediately that $\alpha \geq \beta/2$.

The essence of the proof is very straightforward. If we have $V_\ell = O(2^{-\beta\ell})$ and $C_\ell = O(2^{\gamma\ell})$, then the analysis in Section 1.3 shows that the optimal number of samples N_ℓ on level ℓ is proportional to $2^{-(\beta+\gamma)\ell/2}$, and therefore the cost on level ℓ is proportional to $2^{(\gamma-\beta)\ell/2}$. The result then follows from the requirement that L is chosen so that $(\mathbb{E}[Y] - \mathbb{E}[P])^2 < \varepsilon^2/2$, and the constant of proportionality for N_ℓ is chosen so that $\mathbb{V}[Y] < \varepsilon^2/2$. The proof also has to handle the fact that N_ℓ must be an integer, so the optimal value is rounded up to the nearest integer.

The result of the theorem merits some discussion. In the case $\beta > \gamma$, the dominant computational cost is on the coarsest levels where $C_\ell = O(1)$ and $O(\varepsilon^{-2})$ samples are required to achieve the desired accuracy. This is the standard result for a Monte Carlo approach using i.i.d. samples; to do better would require an alternative approach such as the use of Latin hypercube sampling or quasi-Monte Carlo methods.

In the case $\beta < \gamma$, the dominant computational cost is on the finest levels. Because of condition (i), we have $2^{-\alpha L} = O(\varepsilon)$, and hence $C_L = O(\varepsilon^{-\gamma/\alpha})$. If $\beta = 2\alpha$, which is usually the best that can be achieved since typically $\mathbb{V}[P_\ell - P_{\ell-1}]$ is similar in magnitude to $\mathbb{E}[(P_\ell - P_{\ell-1})^2]$, which is greater than $(\mathbb{E}[P_\ell - P_{\ell-1}])^2$, then the total cost is $O(C_L)$, corresponding to $O(1)$ samples on the finest level, which is the best that can be achieved.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

The dividing case $\beta = \gamma$ is the one for which both the computational effort, and the contributions to the overall variance, are spread approximately evenly across all of the levels; the $(\log \varepsilon)^2$ term corresponds to the L^2 factor in the corresponding discussion at the end of Section 1.3.

One comment on the theorem is that it assumes lots of properties, and then from these determines relatively easily some conclusions for the efficiency of the MLMC approach. In real applications, the tough challenge is in proving that the assumptions are valid, and in particular determining the values of the parameters α, β, γ. Furthermore, the theorem assumes knowledge of the constants c_1, c_2, c_3. In practice, c_1 and c_2 are almost never known, and instead have to be estimated based on empirical estimates of the weak error and the multilevel correction variance.

Collier *et al.* (2014) have developed a modified version of the theorem. Instead of bounding the mean square error, they prefer to use the Central Limit Theorem to construct a confidence interval which bounds $\mathbb{E}[P]$ with a user-prescribed confidence. This exploits the fact that the multilevel correction Y_ℓ on each level is asymptotically normally distributed, and therefore so is Y.

Haji-Ali, Nobile, von Schwerin and Tempone (2014*b*) address two other important points. The first is whether a geometric sequence of levels is the best choice. Their analysis proves that under certain conditions there is negligible benefit in using a non-geometric sequence, but later in Section 2.6 we will discuss other applications in which a non-geometric sequence is appropriate. The second point is that the equal split between the error due to $\mathbb{E}[P_\ell - P_{\ell-1}]$ and the error due to the variance $\mathbb{V}[Y]$ is definitely not optimal. When $\beta > \gamma$, the dominant computational effort is on the coarsest levels; therefore L can be increased significantly with negligible cost, and so it makes sense to allocate most of the error to the variance term, which means that fewer samples will be required. This can give up to a factor 2× improvement in computational cost compared to the equal split used in Giles (2008*a*, 2008*b*).

Equation (2.2) gives the natural choice for the multilevel correction estimator Y_ℓ. However, the multilevel theorem allows for the use of other estimators, provided they satisfy the restriction of condition (ii), which ensures that $\mathbb{E}[Y] = \mathbb{E}[P_L]$. Several examples of this will be given later in this article. Some use slightly different numerical approximations for the coarse and fine paths in SDE simulations, resulting in the estimator

$$Y_\ell = N_\ell^{-1} \sum_{n=1}^{N_\ell} \big(P_\ell^f(\omega^{(n)}) - P_{\ell-1}^c(\omega^{(n)})\big).$$

Provided we maintain the identity

$$\mathbb{E}[P_\ell^f] = \mathbb{E}[P_\ell^c] \tag{2.4}$$

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

so that the expectation on level ℓ is the same for the two approximations, then condition (ii) is satisfied and no additional bias (other than the bias due to the approximation on the finest level) is introduced into the multilevel estimator.

Some others define an antithetic $\omega_a^{(n)}$ with the same distribution as $\omega^{(n)}$, and then use the multilevel estimator

$$Y_\ell = N_\ell^{-1} \sum_{n=1}^{N_\ell} \tfrac{1}{2}\big(P_\ell(\omega^{(n)}) + P_\ell(\omega_a^{(n)})\big) - P_{\ell-1}(\omega^{(n)}).$$

Since $\mathbb{E}[P_\ell(\omega_a^{(n)})] = \mathbb{E}[P_\ell(\omega^{(n)})]$, then again condition (ii) is satisfied.

In each case, the objective in constructing a more complex estimator is to achieve a greatly reduced variance $\mathbb{V}[Y_\ell]$ so that fewer samples are required.

2.2. *Randomized MLMC for unbiased estimation*

A very interesting extension has been introduced in Rhee and Glynn (2012, 2015). Rather than choosing the finest level of simulation L, based on the desired accuracy, and then using the optimal number of samples on each level based on an estimate of the variance, their 'single-term' estimator instead uses N samples in total, and for each sample they perform a simulation on level ℓ with probability p_ℓ.

The estimator is

$$Y = \frac{1}{N} \sum_{n=1}^{N} \frac{1}{p_{\ell^{(n)}}} \big(P_{\ell^{(n)}}^{(n)} - P_{\ell^{(n)}-1}^{(n)}\big)$$

with the level $\ell^{(n)}$ for each sample being selected randomly with the relevant probability. Alternatively, their estimator can be expressed as

$$Y = \sum_{\ell=0}^{\infty} \left(\frac{1}{p_\ell\, N} \sum_{n=1}^{N_\ell} \big(P_\ell^{(n)} - P_{\ell-1}^{(n)}\big) \right),$$

where N_ℓ, the number of samples from level ℓ, is a random variable with

$$\sum_{\ell=0}^{\infty} N_\ell = N, \quad \mathbb{E}[N_\ell] = p_\ell N,$$

and both outer summations can be trivially truncated at the (random) level L beyond which $N_\ell = 0$. Note that in this form it is very similar in appearance to the standard MLMC estimator, which is

$$Y = \sum_{\ell=0}^{L} \left(\frac{1}{N_\ell} \sum_{n=1}^{N_\ell} \big(P_\ell^{(n)} - P_{\ell-1}^{(n)}\big) \right).$$

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

The beauty of their estimator is that it is naturally unbiased, since

$$\begin{aligned}\mathbb{E}[Y] &= \mathbb{E}\left[\frac{1}{p_{\ell'}}(P_{\ell'} - P_{\ell'-1})\right] \\ &= \sum_{\ell=0}^{\infty} p_\ell \, \mathbb{E}\left[\frac{1}{p_{\ell'}}(P_{\ell'} - P_{\ell'-1}) \mid \ell' = \ell\right] \\ &= \sum_{\ell=0}^{\infty} \mathbb{E}[P_\ell - P_{\ell-1}] \\ &= \mathbb{E}[P].\end{aligned}$$

If we define $E_\ell = \mathbb{E}[P_\ell - P_{\ell-1}]$ then the variance of the estimator is

$$\mathbb{V}[Y] = \sum_{\ell=0}^{\infty} \frac{1}{p_\ell}(V_\ell + E_\ell^2) - \left(\sum_{\ell=0}^{\infty} E_\ell\right)^2 \geq \sum_{\ell=0}^{\infty} \frac{1}{p_\ell} V_\ell,$$

due to Jensen's inequality.

The choice of probabilities p_ℓ is crucial. For both the variance and the expected cost to be finite, it is necessary that

$$\sum_{\ell=0}^{\infty} \frac{1}{p_\ell} V_\ell < \infty, \quad \sum_{\ell=0}^{\infty} p_\ell \, C_\ell < \infty.$$

Under the conditions of the MLMC theorem, this is possible when $\beta > \gamma$ by choosing $p_\ell \propto 2^{-(\gamma+\beta)\ell/2}$, so that

$$\frac{1}{p_\ell} V_\ell \propto 2^{-(\beta-\gamma)\ell/2}, \quad p_\ell \, C_\ell \propto 2^{-(\beta-\gamma)\ell/2}.$$

It is not possible when $\beta \leq \gamma$, and for these cases the estimators constructed by Rhee and Glynn (2015) have infinite expected cost.

Provided $\beta > \gamma$, the optimal choice for p_ℓ is

$$p_\ell = \sqrt{V_\ell / C_\ell} \left(\sum_{\ell'=0}^{\infty} \sqrt{V_{\ell'} / C_{\ell'}}\right)^{-1}.$$

If $E_\ell^2 \ll V_\ell$, then the condition that the variance of the estimator is approximately equal to ε^2 gives

$$N \approx \varepsilon^{-2} \sum_{\ell=0}^{\infty} V_\ell / p_\ell \approx \varepsilon^{-2} \left(\sum_{\ell=0}^{\infty} \sqrt{V_\ell \, C_\ell}\right) \left(\sum_{\ell'=0}^{\infty} \sqrt{V_{\ell'} / C_{\ell'}}\right)$$

and therefore the total cost is

$$C = N \sum_{\ell=0}^{\infty} p_\ell \, C_\ell \approx \varepsilon^{-2} \left(\sum_{\ell=0}^{\infty} \sqrt{V_\ell \, C_\ell}\right)^2,$$

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

which is the same as the total cost in equation (1.1), and half the cost of the standard MLMC algorithm in which only half of the MSE 'budget' of ε^2 is allocated to the variance.

Although it is very elegant to have an unbiased estimator when $\beta > \gamma$, the practical benefits may be limited. In this situation, the dominant cost of the standard MLMC algorithm is on the coarsest levels of simulation, and therefore one can increase the value of L substantially at negligible cost, and at the same time allocate almost all of the MSE budget of ε^2 to the variance term in (2.1), leading to approximately the same total cost.

2.3. Multilevel Richardson–Romberg extrapolation

Richardson extrapolation is a very old technique in numerical analysis. Given a numerical approximation P_h based on a discretization parameter h which leads to an error

$$P_h - P = a\, h^\alpha + O(h^{2\alpha}),$$

it follows that

$$P_{2h} - P = a\,(2h)^\alpha + O(h^{2\alpha}),$$

and hence the extrapolated value

$$\widetilde{P} = \frac{2^\alpha}{2^\alpha - 1}\, P_h - \frac{1}{2^\alpha - 1}\, P_{2h}$$

satisfies

$$\widetilde{P} - P = O(h^{2\alpha}).$$

At the suggestion of a reviewer, the author's original MLMC paper (Giles 2008*b*) included results using one level of Richardson extrapolation, both for the multilevel results as well as the standard Monte Carlo method. Multilevel on its own was significantly better than Richardson extrapolation, but the two together were even better.

Lemaire and Pagès have taken this approach much further, and have also performed a comprehensive error analysis (Lemaire and Pagès 2013). Assuming that the weak error has a regular expansion

$$\mathbb{E}[P_\ell] - \mathbb{E}[P] = \sum_{n=1}^{L} a_n 2^{-n\alpha\ell} + O(2^{-\alpha\ell L}),$$

they first determine the unique set of weights w_ℓ, $\ell = 0, 1, \ldots, L$ such that

$$\sum_{\ell=0}^{L} w_\ell = 1, \qquad \sum_{\ell=0}^{L} w_\ell\, 2^{-n\alpha\ell} = 1, \qquad n = 1, \ldots, L,$$

so that

$$\left(\sum_{\ell=0}^{L} w_\ell\, \mathbb{E}[P_\ell]\right) - \mathbb{E}[P] = \sum_{\ell=0}^{L} w_\ell \left(\mathbb{E}[P_\ell] - \mathbb{E}[P]\right) = O(2^{-\alpha L^2}).$$

Next, they rearrange terms to give

$$\sum_{\ell=0}^{L} w_\ell\, \mathbb{E}[P_\ell] = \sum_{\ell=0}^{L} v_\ell\, \mathbb{E}[P_\ell - P_{\ell-1}],$$

where as usual $P_{-1} \equiv 0$, and the coefficients v_ℓ are defined by $w_\ell = v_\ell - v_{\ell+1}$, with $v_{L+1} \equiv 0$, and hence

$$v_\ell = \sum_{\ell'=\ell}^{L} w_{\ell'}.$$

This leads to their multilevel Richardson–Romberg extrapolation estimator,

$$Y = \sum_{\ell=0}^{L} Y_\ell, \quad Y_\ell = N_\ell^{-1} v_\ell \sum_n \left(P_\ell^{(\ell,n)} - P_{\ell-1}^{(\ell,n)}\right).$$

Because the remaining error is $O(2^{-\alpha L^2})$, rather than the usual $O(2^{-\alpha L})$, it is possible to obtain the usual $O(\varepsilon)$ weak error with a value of L which is the square root of the usual value. Hence, in the case $\beta = \gamma$ they prove that the overall cost is reduced to $O(\varepsilon^{-2}|\log \varepsilon|)$, while for $\beta < \gamma$ the cost is reduced much more to $O(\varepsilon^{-2} 2^{(\gamma-\beta)\sqrt{(|\log_2 \varepsilon|/\alpha)}})$. This analysis is supported by numerical results which demonstrate considerable savings (Lemaire and Pagès 2013), and therefore this appears to be a very useful extension to the standard MLMC approach when $\beta \leq \gamma$.

2.4. Multi-index Monte Carlo

Multi-index Monte Carlo (MIMC) is probably the most significant extension of the MLMC methodology (Haji-Ali, Nobile and Tempone 2014*a*). In standard MLMC, there is a one-dimensional set of levels, with a scalar level index ℓ, although in some applications changing ℓ can change more than one aspect of the computation (such as both time-step and spatial discretization in a parabolic SPDE application, or time-step and number of sub-samples in a nested simulation). MIMC generalizes this to ‘levels’ being defined in multiple directions, so that the level ‘index’ $\boldsymbol{\ell}$ is now a vector of integer indices. This is illustrated in Figure 2.1 for a two-dimensional MIMC application.

In MLMC, if we define the backward difference

$$\Delta P_\ell \equiv P_\ell - P_{\ell-1}$$

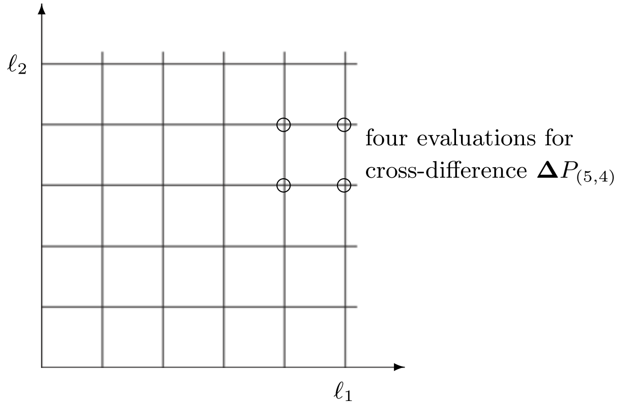

Figure 2.1. 'Levels' in two-dimensional multi-index Monte Carlo application.

with $P_{-1} \equiv 0$, as usual, then the telescoping sum which lies at the heart of MLMC is

$$\mathbb{E}[P] = \sum_{\ell \geq 0} \mathbb{E}[\Delta P_\ell].$$

Generalizing this to D dimensions, Haji-Ali *et al.* (2014*a*) first define a backward difference operator in one particular dimension,

$$\Delta_d P_{\boldsymbol{\ell}} \equiv P_{\boldsymbol{\ell}} - P_{\boldsymbol{\ell} - \boldsymbol{e}_d},$$

where $\boldsymbol{e}_d$ is the unit vector in direction d. They then define the cross-difference

$$\boldsymbol{\Delta} P_{\boldsymbol{\ell}} \equiv \left(\prod_{d=1}^{D} \Delta_d \right) P_{\boldsymbol{\ell}},$$

and the telescoping sum becomes

$$\mathbb{E}[P] = \sum_{\boldsymbol{\ell} \geq 0} \mathbb{E}[\boldsymbol{\Delta} P_{\boldsymbol{\ell}}].$$

As an example, Figure 2.1 marks the four locations at which $P_{\boldsymbol{\ell}}$ must be computed to determine the value of $\boldsymbol{\Delta} P_{(5,4)}$ in the two-dimensional application.

The MIMC theorem formulated by Haji-Ali *et al.* (2014*a*) can be expressed in the following form to match, as closely as possible, the formulation of the MLMC theorem.

Theorem 2.2. If there exist independent estimators $Y_{\boldsymbol{\ell}}$ based on $N_{\boldsymbol{\ell}}$ Monte Carlo samples, each with expected cost $C_{\boldsymbol{\ell}}$ and variance $V_{\boldsymbol{\ell}}$, and positive

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

D-dimensional vectors $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}$, with $\alpha_d \geq \beta_d/2$, and also positive constants c_1, c_2, c_3 such that

(i) $|\mathbb{E}[P_{\boldsymbol{\ell}} - P]| \longrightarrow 0$ as $\min_d \ell_d \longrightarrow \infty$,

(ii) $\mathbb{E}[Y_{\boldsymbol{\ell}}] = \mathbb{E}[\boldsymbol{\Delta} P_{\boldsymbol{\ell}}]$,

(iii) $|\mathbb{E}[Y_{\boldsymbol{\ell}}]| \leq c_1\, 2^{-\boldsymbol{\alpha}\cdot\boldsymbol{\ell}}$,

(iv) $V_{\boldsymbol{\ell}} \leq c_2\, 2^{-\boldsymbol{\beta}\cdot\boldsymbol{\ell}}$,

(v) $C_{\boldsymbol{\ell}} \leq c_3\, 2^{\boldsymbol{\gamma}\cdot\boldsymbol{\ell}}$,

then there exists a positive constant c_4 such that for any $\varepsilon < e^{-1}$ there is a set of levels $\mathcal{L}$, and integers $N_{\boldsymbol{\ell}}$ for which the multilevel estimator

$$Y = \sum_{\boldsymbol{\ell}\in\mathcal{L}} Y_{\boldsymbol{\ell}}$$

has a mean square error with bound

$$\text{MSE} \equiv \mathbb{E}\big[(Y - \mathbb{E}[P])^2\big] < \varepsilon^2,$$

with a computational complexity C with bound

$$\mathbb{E}[C] \leq \begin{cases} c_4\, \varepsilon^{-2} & \eta < 0, \\ c_4\, \varepsilon^{-2}\, |\log\varepsilon|^{e_1} & \eta = 0, \\ c_4\, \varepsilon^{-2-\eta}\, |\log\varepsilon|^{e_2} & \eta > 0, \end{cases}$$

where

$$\eta = \max_d \frac{\gamma_d - \beta_d}{\alpha_d}.$$

When $\alpha_d > \beta_d/2$ for all d, the exponents for the logarithmic terms are

$$e_1 = 2\,D_2, \quad e_2 = (D_2 - 1)\,(2+\eta),$$

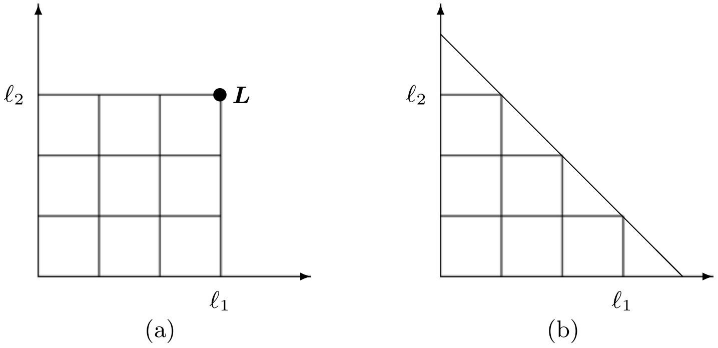

Figure 2.2. Two choices of two-dimensional MIMC summation region $\mathcal{L}$.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

where D_2 is the number of dimensions d for which

$$\frac{\gamma_d - \beta_d}{\alpha_d} = \eta.$$

The form of the exponents is more complicated when $\alpha_d = \beta_d/2$ for some d (Haji-Ali *et al.* 2014*a*).

Notes.

- Condition (i) ensures weak convergence to the correct value.
- Condition (ii) ensures each $Y_{\boldsymbol{\ell}}$ has the correct expected value. This allows for the possibility of using an estimator which is more complicated than the natural choice $Y_{\boldsymbol{\ell}} = \boldsymbol{\Delta} P_{\boldsymbol{\ell}}$, if it will lead to a reduced variance.
- Conditions (iii) and (iv) give the order of convergence of the weak error and the multilevel correction variance, and condition (v) gives the rate at which the cost increases with level.
- The proof of the theorem is significantly harder than for the MLMC theorem.
- There is again the difficulty for each new application of determining $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}$ for conditions (iii)–(v). A relatively simple example will be given later in Section 9.

It might seem natural that the summation region $\mathcal{L}$ should be rectangular, as illustrated in Figure 2.2(a), so that

$$\sum_{\boldsymbol{\ell} \in \mathcal{L}} Y_{\boldsymbol{\ell}} = P_{\boldsymbol{L}}$$

where $\boldsymbol{L}$ is the outermost point on the rectangle. However, Haji-Ali *et al.* (2014*a*) prove that this gives the optimal order of complexity only when $\eta < 0$, and under the additional condition that

$$\sum_d \frac{\gamma_d}{\alpha_d} \leq 2.$$

The optimal choice for $\mathcal{L}$, which yields the complexity bounds given in the theorem, is of the form $\boldsymbol{\ell} \cdot \mathbf{n} \leq L$ for a particular choice of direction vector $\mathbf{n}$ with strictly positive components. In two dimensions, this corresponds to a triangular region, as illustrated in Figure 2.2(b). This is very similar to the use of sparse grid methods in high-dimensional PDE approximations (Bungartz and Griebel 2004), and indeed MIMC can be viewed as a combination of sparse grid methods and Monte Carlo sampling.

The benefits of MIMC over the standard MLMC can be very substantial. They are perhaps best illustrated by an elliptic PDE or SPDE example, in which D corresponds to the number of spatial dimensions. Using the standard MLMC approach, β, the rate of convergence of the multilevel variance,

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

Table 2.1. Dimensionality of Ω, the input space, $\mathcal{U}$, the intermediate solution space, and $\mathcal{P}$, the output space, for some different MLMC applications.

Application	$\dim(\Omega)$	$\dim(\mathcal{U})$	$\dim(\mathcal{P})$
parametric integration	finite	—	∞
SDE with scalar output	∞	∞	1
PDE with scalar output	finite/∞	∞	1
SDE with CDF output	∞	∞	∞

will usually be independent of D, but γ, the rate of increase in the computational cost, will increase at least linearly with D. Therefore, in a high enough dimension we will have $\beta \leq \gamma$ and therefore the overall computational complexity will be less (often much less) than the optimal $O(\varepsilon^{-2})$. However, using MIMC we can have $\beta_d > \gamma_d$ in each direction, and therefore obtain the optimal complexity.

Hence, in the same way that sparse grids offer the possibility of dimension-independent complexity for deterministic PDE applications, MIMC offers the possibility of dimension-independent complexity for SPDEs and other high-dimensional stochastic applications.

2.5. Multi-dimensional output functionals

In Monte Carlo simulation, we start with a sample ω from the random input space, often compute an intermediate solution U from an approximation to an SDE or SPDE, and then determine the final output quantity P.

$$\omega \longrightarrow U \longrightarrow P$$

So far, the discussion in this article has focused on the idea of P being a scalar output, but this is not necessarily always the case. In some applications there may be more than one scalar output of interest, and in other applications P may be infinite-dimensional.

Some of the possibilities are listed in Table 2.1. In Heinrich's original multilevel research on parametric integration (Heinrich 1998, Heinrich and Sindambiwe 1999, Heinrich 2000, Heinrich 2001), the sample ω was from a finite-dimensional unit hypercube, there was no intermediate solution, and the output was an infinite-dimensional function of a parameter λ. In the author's initial research (Giles 2008*a*, 2008*b*), ω represented an infinite-dimensional input Brownian path, U was the solution of an SDE, and P was a scalar functional of that solution U. In the simplest PDE applications, one might have a finite set of parameters which specify uncertain initial or boundary data, and a single scalar output, so only the intermediate PDE

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

solution is infinite-dimensional. Finally, new research (Giles, Nagapetyan and Ritter 2015*b*) addresses the problem of approximating the cumulative distribution function (CDF) of exit times associated with the solution of an SDE; in this case, all three spaces are infinite-dimensional.

The extension of the MLMC theory to a finite set of outputs is very natural. The standard MLMC theorem will apply to each one of the outputs. If there is a desired accuracy ε_m for the mth output, then a simple approach proposed by Tigran Nagapetyan[1] is to define

$$V_\ell \equiv \max_m \frac{V_{\ell,m}}{\varepsilon_m^2},$$

where $V_{\ell,m}$ is the variance of the multilevel estimator on level ℓ for the mth output. Enforcing the constraint

$$\sum_{\ell=0}^{L} N_\ell^{-1} V_\ell \leq \tfrac{1}{2}$$

is then sufficient to ensure that all of the individual variance constraints are satisfied, and the standard Lagrange multiplier approach outlined in Section 1.3 can be used to determine the optimal number of samples to use on each level.

The extension of the theory to infinite-dimensional outputs requires the definition of an appropriate norm $\|\cdot\|$, and then to a large extent the theory carries over by making the substitutions

$$\begin{aligned} |\mathbb{E}[P_\ell] - \mathbb{E}[P]| &\longrightarrow \|\mathbb{E}[P_\ell] - \mathbb{E}[P]\|, \\ \mathbb{V}[P_\ell - P_{\ell-1}] &\longrightarrow \mathbb{E}\big[\|P_\ell - P_{\ell-1} - \mathbb{E}[P_\ell - P_{\ell-1}]\|^2\big]. \end{aligned}$$

One slight complication is that if a and b are two independent scalar random variables with zero mean, then

$$\mathbb{E}\big[(a+b)^2\big] = \mathbb{E}\big[a^2\big] + \mathbb{E}\big[b^2\big].$$

When using the 2-norm, this extends to independent random vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, each with zero mean, since

$$\mathbb{E}\big[\|\boldsymbol{a}+\boldsymbol{b}\|^2\big] = \mathbb{E}\big[\|\boldsymbol{a}\|^2\big] + \mathbb{E}\big[\|\boldsymbol{b}\|^2\big],$$

and similarly to random functions with a 2-norm based on an inner product. However this does not necessarily apply for other norms, and so the variance for the combined multilevel estimator cannot necessarily be expressed in the usual form as

$$\sum_\ell N_\ell^{-1} V_\ell, \qquad V_\ell \equiv \mathbb{E}\big[\|P_\ell - P_{\ell-1} - \mathbb{E}[P_\ell - P_{\ell-1}]\|^2\big].$$

[1] Personal communication.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

Nevertheless, the theory can be extended to the use of other norms by using results from Banach space theory for the sums of independent random variables (Ledoux and Talagrand 1991, Heinrich 1998), and in recent research Daun and Heinrich have addressed the challenges of parametric integration in infinite-dimensional Banach spaces (Daun and Heinrich 2013, 2014*a*, 2014*b*).

2.6. Non-geometric MLMC

It is very important to repeat, and emphasize, the fact that MLMC does not require the use of a geometric sequence of grids. In many applications, and most of the existing literature, a geometric sequence will be the appropriate choice, but there are other applications for which it is not. The only assumptions in the analysis of the two-level method in Section 1.2 and the multilevel method in Section 1.3 are that the accuracy and cost increase with level, and the variance of the correction term decreases with level.

When there is no underlying structure to suggest that a geometric sequence is appropriate, a question which arises is how to select the levels. Suppose that one starts with a large collection of potential levels; how does one decide which subset to use?

One approach developed by Vidal-Codina, Nguyen, Giles and Peraire (2014) is to first use a number of trial samples to obtain an estimate of $V_\ell = \mathbb{V}[P_\ell]$ and $V_{\ell_1,\ell_2} \equiv \mathbb{V}[P_{\ell_2} - P_{\ell_1}]$ for all pairs (ℓ_1, ℓ_2), with $\ell_1 < \ell_2$, with associated costs C_ℓ and C_{ℓ_1,ℓ_2}. For a particular ordered subset of levels, $\{\ell_1, \ell_2, \dots, \ell_M\}$, with fixed $\ell_M = L$ to achieve a user-specified accuracy, following the analysis in Section 1.3 which led to (1.1), we obtain the cost

$$C(\ell_1, \ell_2, \dots, \ell_M) = \varepsilon^{-2}\left(\sqrt{V_{\ell_0}\, C_{\ell_0}} + \sum_{m=1}^{M} \sqrt{V_{\ell_m,\ell_{m-1}}\, C_{\ell_m,\ell_{m-1}}}\right)^2.$$

If the maximum number of levels is not too large, it is possible to perform an exhaustive search to find the optimal subset $\{\ell_1, \ell_2, \dots, \ell_M\}$ which minimizes this cost.

Alternatively, suppose that we have a multilevel Monte Carlo implementation with a number of levels $\ell = 0, 1, \dots, L$. Looking in particular at one level ℓ, is it computationally more efficient to keep that level, or would it be better to drop it and jump straight from level $\ell - 1$ to $\ell + 1$?

If we keep level ℓ then the contribution to the overall multilevel estimator due to samples involving level ℓ is

$$\frac{1}{N_\ell} \sum_{n=1}^{N_\ell} (P_\ell^{(n)} - P_{\ell-1}^{(n)}) + \frac{1}{N_{\ell+1}} \sum_{n=1}^{N_{\ell+1}} (P_{\ell+1}^{(n)} - P_\ell^{(n)}).$$

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

The usual multilevel analysis shows that for optimality $N_\ell \propto \sqrt{V_\ell/C_\ell}$, where $V_\ell \equiv \mathbb{V}[P_\ell - P_{\ell-1}]$ and C_ℓ is the cost of computing a single sample of $P_\ell - P_{\ell-1}$. Hence we have

$$N_\ell = N_{\ell+1}\sqrt{\frac{V_\ell}{V_{\ell+1}}\,\frac{C_{\ell+1}}{C_\ell}}.$$

The combined variance from the two terms is

$$\frac{1}{N_\ell}V_\ell + \frac{1}{N_{\ell+1}}V_{\ell+1} = \frac{V_{\ell+1}}{N_{\ell+1}}\left(1+\sqrt{\frac{V_\ell\,C_\ell}{V_{\ell+1}\,C_{\ell+1}}}\right),$$

and the combined cost is

$$N_\ell\,C_\ell + N_{\ell+1}\,C_{\ell+1} = N_{\ell+1}\,C_{\ell+1}\left(1+\sqrt{\frac{V_\ell\,C_\ell}{V_{\ell+1}\,C_{\ell+1}}}\right),$$

and so the product of the combined variance and combined cost is

$$V_{\ell+1}\,C_{\ell+1}\left(1+\sqrt{\frac{V_\ell\,C_\ell}{V_{\ell+1}\,C_{\ell+1}}}\right)^2.$$

On the other hand, if we drop level ℓ and jump straight from level $\ell-1$ to $\ell+1$, then the replacement estimator is

$$\frac{1}{\widetilde{N}_{\ell+1}}\sum_{n=1}^{\widetilde{N}_\ell}(P^{(n)}_{\ell+1} - P^{(n)}_{\ell-1}).$$

Its variance is $\widetilde{V}_{\ell+1}/\widetilde{N}_{\ell+1}$, where $\widetilde{V}_{\ell+1} \equiv \mathbb{V}[P_{\ell+1} - P_{\ell-1}]$, and its cost is $\widetilde{N}_{\ell+1}\widetilde{C}_{\ell+1}$, where $\widetilde{C}_{\ell+1}$ is the cost of a single sample of $P_{\ell+1} - P_{\ell-1}$. Hence, the product of the variance and the cost is

$$\widetilde{V}_{\ell+1}\;\widetilde{C}_{\ell+1}.$$

The question now is whether

$$\widetilde{V}_{\ell+1}\,\widetilde{C}_{\ell+1} < V_{\ell+1}\,C_{\ell+1}\left(1+\sqrt{\frac{V_\ell\,C_\ell}{V_{\ell+1}\,C_{\ell+1}}}\right)^2. \tag{2.5}$$

If this test is true, then it is best to drop level ℓ, because for a fixed computational cost this will deliver the lower variance, or for a fixed variance it can be achieved at a lower computational cost.

The main contribution to both $C_{\ell+1}$ and $\widetilde{C}_{\ell+1}$ comes from the level $\ell+1$ simulation which produces the output $P_{\ell+1}$, so we will assume from here on that $\widetilde{C}_{\ell+1} \approx C_{\ell+1}$. The cost ratio $C_\ell/C_{\ell+1}$ is usually known *a priori*, and the variance ratio $V_\ell/V_{\ell+1}$ is often known asymptotically as $\ell \to \infty$, or can be estimated empirically. The key question is how $\widetilde{V}_{\ell+1}$ compares to $V_{\ell+1}$.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

This could also be quantified empirically, by generating a few Monte Carlo samples for $P_{\ell+1}-P_{\ell-1}$. Alternatively, we can perform some analysis to gain some insight into the question.

Using the standard result for the variance of a sum of two random variables, we have

$$\widetilde{V}_{\ell+1} = V_{\ell+1} + 2\rho\sqrt{V_\ell\, V_{\ell+1}} + V_\ell,$$

where ρ is the correlation between $P_{\ell+1} - P_\ell$ and $P_\ell - P_{\ell-1}$. We can now consider two extremes, $\rho = 1$ and $\rho = 0$.

If $\rho = 1$, and so we have perfect correlation between the increments at different levels, then

$$\widetilde{V}_{\ell+1} = V_{\ell+1}\left(1+\sqrt{\frac{V_\ell}{V_{\ell+1}}}\right)^2.$$

Since $C_\ell < C_{\ell+1}$, it follows that the test (2.5) is never satisfied, and so it is best to retain level ℓ. Near-perfect correlation is likely to arise in PDE applications with finite-dimensional uncertainty in the initial data, boundary data, or PDE coefficients. In this case, the increments $P_\ell - P_{\ell-1}$ obtained in going from one level to the next come from the progressive elimination of truncation error due to the discretization of the PDE.

On the other hand, if $\rho = 0$, and so the increments at different levels are independent, then $\widetilde{V}_{\ell+1} = V_{\ell+1} + V_\ell$ and so we drop level ℓ if

$$1 + \frac{V_\ell}{V_{\ell+1}} < \left(1+\sqrt{\frac{V_\ell\, C_\ell}{V_{\ell+1}\, C_{\ell+1}}}\right)^2.$$

2.7. MLQMC

Giles and Waterhouse (2009) developed a variant of MLMC which uses quasi-Monte Carlo samples instead of independent Monte Carlo samples. This was based on extensible rank-1 lattices developed at UNSW (Dick, Pillichshammer and Waterhouse 2007). QMC is known to be most effective for low-dimensional applications, and the numerical results were very encouraging for SDE applications in which the dominant computational cost was on the coarsest levels of resolution. However, there was no supporting theory for this research.

More recently, there has been considerable research on the theoretical foundations for multilevel QMC (MLQMC) methods; see Niu, Hickernell, Müller-Gronbach and Ritter (2010), Kuo, Schwab and Sloan (2015), Dick *et al.* (2013), Baldeaux and Gnewuch (2014) and Dick and Gnewuch (2014*a*, 2014*b*). These theoretical developments are very encouraging, showing that under certain conditions they lead to multilevel methods with a complexity which is $O(\varepsilon^{-p})$ with $p < 2$.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

3. MLMC implementation

3.1. MLMC algorithm

Based on the theory in Section 2.1, the geometric MLMC algorithm used for the numerical tests in this paper is given in Algorithm 1.

Algorithm 1 MLMC algorithm

```
start with $L = 2$, and initial target of $N_0$ samples on levels $\ell = 0, 1, 2$
while extra samples need to be evaluated do
    evaluate extra samples on each level
    compute/update estimates for $V_\ell, \ell = 0, \ldots, L$
    define optimal $N_\ell, \ell = 0, \ldots, L$
    test for weak convergence
    if not converged, set $L := L + 1$, and initialize target $N_L$
end while
```

In the above algorithm, the equation for the optimal N_ℓ is

$$N_\ell = \left\lceil 2\,\varepsilon^{-2}\,\sqrt{V_\ell/C_\ell}\,\left(\sum_{\ell=0}^{L}\sqrt{V_\ell\,C_\ell}\right)\right\rceil, \tag{3.1}$$

where V_ℓ is the estimated variance, and C_ℓ is the cost of an individual sample on level ℓ. This ensures that the estimated variance of the combined multilevel estimator is less than $\varepsilon^2/2$.

The test for weak convergence tries to ensure that $|\mathbb{E}[P - P_L]| < \varepsilon/\sqrt{2}$, to achieve an MSE which is less than ε^2, with ε being a user-specified RMS accuracy. If $\mathbb{E}[P_\ell - P_{\ell-1}] \ \propto\ 2^{-\alpha\ell}$, then the remaining error is

$$\mathbb{E}[P - P_L] = \sum_{\ell=L+1}^{\infty} \mathbb{E}[P_\ell - P_{\ell-1}] = \mathbb{E}[P_L - P_{L-1}] \;/\; (2^\alpha - 1).$$

This leads to the convergence test $|\mathbb{E}[P_L - P_{L-1}]|/(2^\alpha - 1) < \varepsilon/\sqrt{2}$, but for robustness, we extend this check to extrapolate from the previous two data points $\mathbb{E}[P_{L-1} - P_{L-2}]$, $\mathbb{E}[P_{L-2} - P_{L-3}]$, and take the maximum over all three as the estimated remaining error.

It is important to note that this algorithm is heuristic; it is not guaranteed to achieve an MSE error which is less than ε^2. The main MLMC theorem in Section 2.1 does provide a guarantee, but the conditions of the theorem assume *a priori* knowledge of the constants c_1 and c_2 governing the weak convergence and the variance convergence as $\ell \to \infty$. These two constants are in effect being estimated in the numerical algorithm described above.

In addition, the accuracy of the variance estimate at each level depends on the size of the sample set. If it is small, then this variance estimate may

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

be poor; we will see later that the implementation sets a minimum estimate by extrapolation from coarser levels with more samples.

3.2. Overview of MATLAB implementation

The results to be presented later all use the same two routines written in MATLAB.

- `mlmc.m`: driver code which performs the MLMC calculation using an application-specific routine to compute $\sum_n (P_\ell^{(n)} - P_{\ell-1}^{(n)})^p$ for $p = 1, 2$ and a specified number of independent samples.
- `mlmc_test.m`: a program which perform various tests and then calls `mlmc.m` to perform a number of MLMC calculations.

These two routines, and all of the application-specific routines, can be downloaded from Giles (2014).

3.3. MLMC test routine

`mlmc_test.m` first performs a set of calculations using a fixed number of samples on each level of resolution, and produces four plots:

- $\log_2(V_\ell)$ versus level ℓ,
- $\log_2(|\mathbb{E}[P_\ell - P_{\ell-1}]|)$ versus level ℓ,
- consistency check versus level,
- kurtosis versus level.

The last two of these need some explanation. If a, b, c are estimates for $\mathbb{E}[P_{\ell-1}^f]$, $\mathbb{E}[P_\ell^f]$, $\mathbb{E}[Y_\ell]$, respectively, then it should be true that $a - b + c \approx 0$. The consistency check verifies that this is true, to within the accuracy one would expect due to Monte Carlo sampling error. This is particularly important when $P_\ell^f \neq P_\ell^c$ because one is using different approximations on the coarse and fine levels.

Since

$$\sqrt{\mathbb{V}[a - b + c]} \leq \sqrt{\mathbb{V}[a]} + \sqrt{\mathbb{V}[b]} + \sqrt{\mathbb{V}[c]},$$

it computes and plots the ratio

$$\frac{|a - b + c|}{3(\sqrt{V_a} + \sqrt{V_b} + \sqrt{V_c})},$$

where V_a, V_b, V_c are empirical estimates for the variances of a, b, c. The probability of this ratio being greater than unity is less than 0.3%. Hence, if it is, it indicates a likely programming error, or a mathematical error in the formulation which violates the crucial identity (2.4).

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

The MLMC approach needs a good estimate for $V_\ell = \mathbb{V}[Y_\ell]$, but how many samples are needed for this? As few as 10 may be sufficient in many cases for a rough estimate, but many more are needed when there are rare outliers. When the number of samples N is large, the standard deviation of the sample variance for a random variable X with zero mean is approximately $\sqrt{(\kappa-1)/N}\ \mathbb{E}[X^2]$, where the kurtosis κ is defined as $\kappa = \mathbb{E}[X^4]/\mathbb{E}[X^2])^2$. Hence $O(\kappa)$ samples are required to obtain a reasonable estimate for the variance. As well as plotting the kurtosis κ_ℓ, `mlmc_test.m` will give a warning if κ_ℓ is very large, as this is an indication that the empirical variance estimate may be poor.

An extreme, but important, example is when P always takes the value 0 or 1. In this case we have

$$X \equiv P_\ell - P_{\ell-1} = \begin{cases} 1 & \text{probability } p, \\ -1 & \text{probability } q, \\ 0 & \text{probability } 1-p-q. \end{cases}$$

If $p, q \ll 1$, then $\mathbb{E}[X] \approx 0$, and $\kappa \approx (p+q)^{-1} \gg 1$. Therefore, many samples are required for a good estimate of V_ℓ; otherwise, we may get all $X^{(n)} = 0$, which will give an estimated variance of zero. What is more, the kurtosis will become worse as $\ell \to \infty$ since $p, q \to 0$ due to weak convergence.

After performing these initial tests, `mlmc_test.m` then uses the driver routine `mlmc.m` to perform the MLMC computation for a number of values of ε, and plots the results. These plots will be explained later when the first results are presented.

3.4. MLMC driver routine

The first part of the documentation section at the top of `mlmc.m` is fairly self-explanatory.

```
% function [P, Nl] = mlmc(N0,eps,mlmc_l, alpha,beta,gamma)
%
% P     = value
% Nl    = number of samples at each level
% N0    = initial number of samples on levels 0,1,2
% eps   = desired accuracy (RMS error)
% alpha  -> weak error is  O(2^{-alpha*l})
% beta   -> variance is    O(2^{-beta*l})
% gamma  -> sample cost is O(2^{gamma*l})
%
% if alpha, beta are not positive then they will be estimated
```

The user must supply an application-specific function to perform the calculations to compute Y_ℓ on level ℓ. This function has to conform to the following template.

```
% mlmc_l = application-specific function for level l estimator
%          which must conform to the following template:
%
% function sums = mlmc_l(l,N)
% inputs:  l = level
%          N = number of paths
% output: sums(1) = sum(Y)
%         sums(2) = sum(Y.^2)
%         where Y are iid samples with expected value:
%         E[P_0] on level 0
%         E[P_l - P_{l-1}] on level l>0
```

The first part of the code initializes various parameters, and then uses the application-specific routine mlmc_l to generate the initial N0 samples on each of the first three levels.

```
function [P, Nl] = mlmc(N0,eps,mlmc_l, alpha_0,beta_0,gamma)
  alpha = max(0, alpha_0);
  beta  = max(0, beta_0);

  L              = 2;
  Nl(1:3)        = 0;
  suml(1:2,1:3) = 0;
  dNl(1:3)       = N0;

% keep looping while additional samples need to be computed

  while sum(dNl) > 0

% update sample sums on each level

    for l=0:L
      if dNl(l+1) > 0
        sums        = feval(mlmc_l,l,dNl(l+1));
        Nl(l+1)     = Nl(l+1) + dNl(l+1);
        suml(1,l+1) = suml(1,l+1) + sums(1);
        suml(2,l+1) = suml(2,l+1) + sums(2);
      end
    end
```

The next part computes (or in later passes updates) the estimates for $m_\ell \equiv |\mathbb{E}[Y_\ell]|$, and $V_\ell \equiv \mathbb{V}[Y_\ell]$. If the mean and variance are decaying as expected, one would expect $m_\ell = 2^{-\alpha}\, m_{\ell-1}$, $V_\ell = 2^{-\beta}\, V_{\ell-1}$. To avoid possible problems from high kurtosis generating poor empirical estimates, the

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

estimates for m_ℓ and V_ℓ are not allowed to decrease by more than factor 1/2 relative to this anticipated value.[2]

```
% compute absolute average and variance

    ml = abs(   suml(1,:)./Nl);
    Vl = max(0, suml(2,:)./Nl - ml.^2);

    for l = 3:L+1
      ml(l) = max(ml(l), 0.5*ml(l-1)/2^alpha);
      Vl(l) = max(Vl(l), 0.5*Vl(l-1)/2^beta);
    end
```

If the user has not supplied the values for α and/or β, they are now estimated by linear regression.

```
% use linear regression to estimate alpha, beta if not given

    if alpha_0 <= 0
      x     = [(1:L)' ones(L,1)] \ log2(ml(2:end))';
      alpha = max(0.5,-x(1));
    end

    if beta_0 <= 0
      x    = [(1:L)' ones(L,1)] \ log2(Vl(2:end))';
      beta = max(0.5,-x(1));
    end
```

Next, the optimal number of samples on each level is computed, which in turn gives the number of additional samples which will need to be generated in the next pass.

```
% set optimal number of additional samples

    Cl  = 2.^(gamma*(0:L));
    Ns  = ceil(2 * sqrt(Vl./Cl) * sum(sqrt(Vl.*Cl)) / eps^2);
    dNl = max(0, Ns-Nl);
```

In the final part of the loop, we test for weak convergence, using the algorithm described in Section 3.1. If a new level is added, its variance is estimated by extrapolation, and the optimal number of samples is updated.

[2] Haji-Ali *et al.* (2014*a*) use an alternative Bayesian approach to address this same problem of estimating the variance.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

```
% if (almost) converged, estimate remaining error and decide
% whether a new level is required

    if sum( dNl > 0.01*Nl ) == 0
      range = -2:0;
      rem = max(ml(L+1+range).*2.^(alpha*range)) / (2^alpha - 1);

      if rem > eps/sqrt(2)
        L       = L+1;
        Vl(L+1) = Vl(L) / 2^beta;
        Nl(L+1) = 0;
        suml(1:4,L+1) = 0;

        Cl  = 2.^(gamma*(0:L));
        Ns  = ceil(2 * sqrt(Vl./Cl) * sum(sqrt(Vl.*Cl)) / eps^2);
        dNl = max(0, Ns-Nl);
      end
    end
  end
```

The final step after the completion of the main loop is to compute the combined multilevel estimator.

```
% finally, evaluate multilevel estimator

  P = sum(suml(1,:)./Nl);
```

3.5. *MLQMC algorithm*

We conclude this chapter with the description of an MLQMC algorithm which is based on that developed by Giles and Waterhouse (2009), but updated to bring it more in line with the MLMC algorithm in Section 3.1.

The key change in MLQMC is that N_ℓ is now the size of a set of QMC points used on level ℓ. This set of points is not constructed randomly and independently but instead constructed very carefully, using well-established QMC techniques such as rank-1 lattices (Dick *et al.* 2007) or Sobol sequences (Joe and Kuo 2008) to provide a relatively uniform coverage of a unit hypercube integration region. In the best cases, this results in the approximate numerical integration error being $O(N_\ell^{-1})$ rather than the usual $O(N_\ell^{-1/2})$ error which comes from Monte Carlo sampling.

Using just one set of N_ℓ points gives good accuracy, but no confidence interval. To regain a confidence interval one uses randomized QMC in which the set of points is gives a random shift (for rank-1 lattice rules) or a digital scrambling (for Sobol sequences). Using 32 sets of points, each collectively randomized, yields 32 set averages for the quantity of interest, Y_ℓ, and from these 32 random independent values the variance of their average, V_ℓ, can be estimated in the usual way.

Since V_ℓ is now defined to be the variance of the average, for a given number of levels L, the aim is to choose the N_ℓ in a way which ensures that

$$\sum_{\ell=0}^{L} V_\ell \leq \varepsilon^2/2. \tag{3.2}$$

How can this be achieved most efficiently? Many QMC methods work naturally with N_ℓ as a power of 2. Doubling N_ℓ will usually eliminate a large fraction of the variance, so the greatest reduction in total variance relative to the additional computational effort is achieved by doubling N_ℓ on the level ℓ^* given by

$$\ell^* = \arg\min_\ell \frac{V_\ell}{N_\ell \, C_\ell}. \tag{3.3}$$

Based on this, and using the same test for weak convergence as in the MLMC algorithm, the MLQMC algorithm is given in Algorithm 2.

Algorithm 2 MLQMC algorithm

```
start with $L = 2$, and an initial set size $N_\ell = 1$ on levels $\ell = 0, 1, 2$
while not converged do
    evaluate 32 set averages on any level with new/changed values of $N_\ell$
    compute corresponding new/changed estimates for $V_\ell$
    test whether total variance condition (3.2) is satisfied
    if total variance still too big then
        determine $\ell^*$ defined by (3.3) and double $N_{\ell^*}$
    else
        test for weak convergence
        if not converged then
            set $L := L + 1$, and initialize $N_L = 1$
        end if
    end if
end while
```

4. Introduction to applications

The remainder of the article presents and discusses a large number of different applications using MLMC methods. In part, the large number is to illustrate the flexibility and generality of the approach, and stimulate ideas on how it could be used for other applications which the reader may be interested in, but the variety of applications is also chosen to illustrate different responses to the following questions.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

(1) How are the levels defined?

This is often quite simple, but in some cases, for example when using adaptive time-stepping, it is not so obvious.

(2) How are the coarse and fine samples coupled when computing the difference $P_\ell(\omega) - P_{\ell-1}(\omega)$?

What does it even mean for $P_\ell(\omega)$ and $P_{\ell-1}(\omega)$ to correspond to the same sample ω? In some cases, it is very simple, but in other cases, such as for the continuous-time Markov chains, it requires a key insight to construct a suitable coupling.

(3) Are there problems with discontinuous outputs?

When the output function is a discontinuous function of the intermediate solution U, there is the possibility that a small difference between the intermediate solutions for the coarse and fine samples leads to an $O(1)$ value for $P_\ell - P_{\ell-1}$. This usually leads to a larger variance, and hence a lower value for β. Addressing this problem is a critical issue, and we will consider a variety of different techniques. Many of these exploit the flexibility of slightly different approximations for the coarse and fine samples, provided always that the identity (2.4) is respected, so that the telescoping sum is still valid.

(4) Is there scope for constructing even more efficient multilevel estimators?

As with the techniques for handling discontinuous output functions, there can be ways of exploiting the flexibility of different coarse and fine approximations to achieve a higher rate of variance convergence than one would otherwise expect due to the rate of strong convergence of the intermediate solution.

(5) What is the current status on supporting numerical analysis to explain the observed behaviour of numerical algorithms?

For many applications this is the greatest challenge, but great progress has been made in the past five years.

5. SDEs

5.1. *Euler–Maruyama discretization*

The original multilevel path simulation paper (Giles 2008*b*) treated stochastic differential equations

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

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

using the simple Euler–Maruyama discretization with a uniform time-step h and Brownian increments ΔW_n,

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

The multilevel Monte Carlo implementation is very simple. On level ℓ, the uniform time-step is taken to be $h_\ell = h_0\, M^{\ell}$, for some integer M. The time-step h_0 on the coarsest level is often taken to be the interval length T, so that there is just one time-step for the entire interval, but this is not required, and in some applications using such a large time-step may lead to numerical results which are so inaccurate that they are not helpful, that is, they are such a poor control variate for the finer approximations that they do not lead to a reduction in the variance. In such cases, it would be better to start with a smaller value for h_0.

The multilevel coupling is achieved by using the same underlying driving Brownian path for the coarse and fine paths; this is accomplished by summing the Brownian increments for the fine path time-steps to obtain the Brownian increments for the coarse time-steps. The multilevel estimator is then the natural one defined in (2.2), with the specific payoff approximation P_ℓ depending on the particular application.

Provided the SDE satisfies the usual conditions (see Theorem 10.2.2 in Kloeden and Platen 1992), the strong error for the Euler discretization with time-step h is $O(h^{1/2})$, so that

$$\mathbb{E}\big[\|S - \widehat{S}\|^2\big] = O(h).$$

For Lipschitz payoff functions P (such as European, Asian and lookback options in finance) for which

$$|P(S_1) - P(S_2)| \le K \|S_1 - S_2\|,$$

we have

$$\mathbb{V}[P - P_\ell] \le \mathbb{E}\big[(P - P_\ell)^2\big] \le K^2\, \mathbb{E}\big[\|S - \widehat{S}_\ell\|^2\big],$$

and

$$V_\ell \equiv \mathbb{V}[P_\ell - P_{\ell-1}] \le 2\big(\mathbb{V}[P - P_\ell] + \mathbb{V}[P - P_{\ell-1}]\big),$$

and hence $V_\ell = O(h_\ell)$.

If $h_\ell = 4^{-\ell} h_0$, as in the numerical examples in Giles (2008*b*), then this gives $\alpha = 2$, $\beta = 4$ and $\gamma = 2$. Alternatively, if $h_\ell = 2^{-\ell} h_0$ with twice as many time-steps on each successive level, as used in the numerical examples in this article, then $\alpha = 1$, $\beta = 2$ and $\gamma = 1$. In either case, Theorem 2.1 gives the complexity to achieve an RMS error of ε to be $O(\varepsilon^{-2}(\log \varepsilon)^2)$, which is near-optimal, because Creutzig, Dereich, Müller-Gronbach and Ritter (2009) prove an $O(\varepsilon^{-2})$ lower bound for the complexity. The same paper also proves that MLMC achieves an $O(\varepsilon^{-2}(\log \varepsilon)^2)$ worst-case optimal complexity for a class of Lipschitz path-dependent functions.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

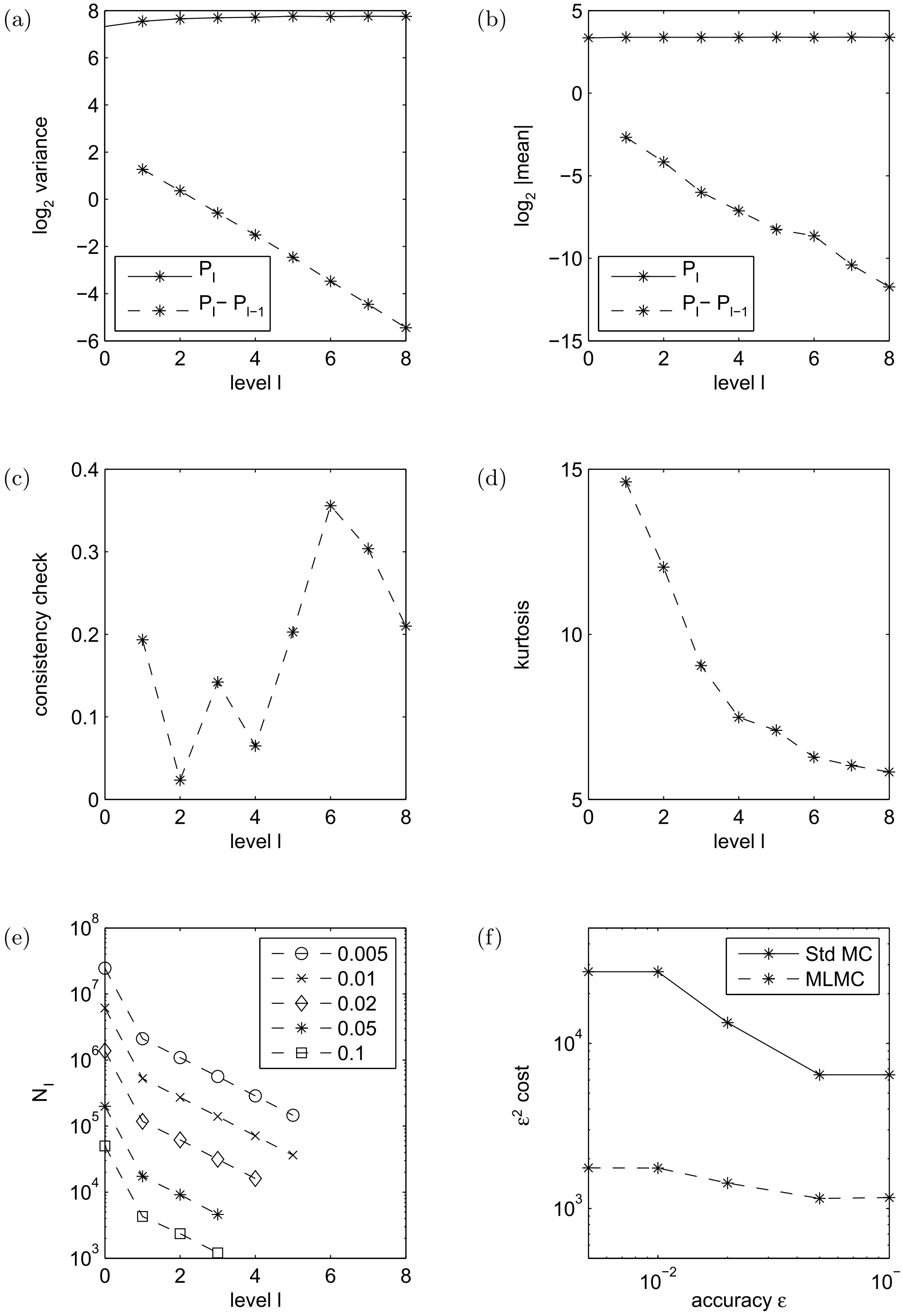

Figure 5.1. Numerical results for a European call option using the Euler–Maruyama discretization of the GBM SDE.

Figure 5.1 shows results for a financial call option with a single underlying asset satisfying the geometric Brownian motion SDE,

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

and payoff $P(S) \equiv \exp(-rT)\,\max(S_T - K, 0)$ with $r = 0.05, \sigma = 0.2, T = 1, S_0 = 100, K = 100$. The full code is provided in Giles (2014).

Plots (a) and (b) show the variance and absolute mean value for both P_ℓ and $P_\ell - P_{\ell-1}$. These are based on an initial sample of 10^5 paths. The line for $\log_2 \mathbb{V}[P_\ell - P_{\ell-1}]$ in plot (a) has a slope of approximately -1, corresponding to a variance proportional to h_ℓ, as expected. The line for $\log_2 |\mathbb{E}[P_\ell - P_{\ell-1}]|$ also has a slope of approximately -1, implying an $O(h_\ell)$ weak convergence.

Plots (c) and (d) show the consistency check and kurtosis, as discussed in Section 3.3. These indicate that there is no problem with the consistency, and the kurtosis is actually improving slightly with level.

Plots (e) and (f) have the results from five separate MLMC calculations, each with a different specified accuracy ε to be achieved. Plot (e) shows the number of samples which is used for each level of the MLMC computation. The number of levels increases as ε decreases, as described in Section 3.1, to achieve the required weak error. The number of samples also varies with level, with many more on the coarsest level with just one time-step. On the finer levels, the optimal allocation of computational effort leads to $N_\ell \propto \sqrt{V_\ell / C_\ell} \approx 2^{-\ell}$.

Plot (f) displays the total computational cost C, multiplied by ε^2. If C were approximately proportional to ε^{-2} then $\varepsilon^2\,C$ would be approximately constant. Indeed, this is what is seen, with the slight increase as ε decreases being due to the $|\log \varepsilon|^2$ term in Theorem 2.1. For comparison, plot (f) also displays the cost using the standard Monte Carlo algorithm on the finest level used by the MLMC algorithm. The cost ratio is between 5 and 12, illustrating the benefits provided by the MLMC algorithm.

Figure 5.2 illustrates the problem with discontinuous payoff functions. The underlying SDE is exactly the same, but the payoff is a digital option with payoff $P(S) \equiv 10\,\exp(-rT)\,H(S_T - K)$, where $H(x)$ is the Heaviside step function. On the finer levels, most samples have fine and coarse paths which end on the same side of the strike K, and hence give $P_\ell - P_{\ell-1} = 0$. However, noting that the strong error is $O(h_\ell^{1/2})$, and there is a bounded density of paths terminating in the neighbourhood of K, there is therefore an $O(h_\ell^{1/2})$ fraction of the samples with the coarse and fine paths on either side of the strike, with $P_\ell - P_{\ell-1} = \pm 1$. This gives $V_\ell = O(h^{1/2})$, and this lower rate of convergence is clearly visible in plot (a). Furthermore, $\mathbb{E}[(P_\ell - P_{\ell-1})^4] = O(h_\ell^{1/2})$ and so the kurtosis is $O(h_\ell^{-1/2})$; this increase in kurtosis is obvious in plot (d), which illustrates why this is a helpful quantity to plot. Consequently, this application has $\alpha = 1$, $\beta = 1/2$, $\gamma = 1$, leading

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

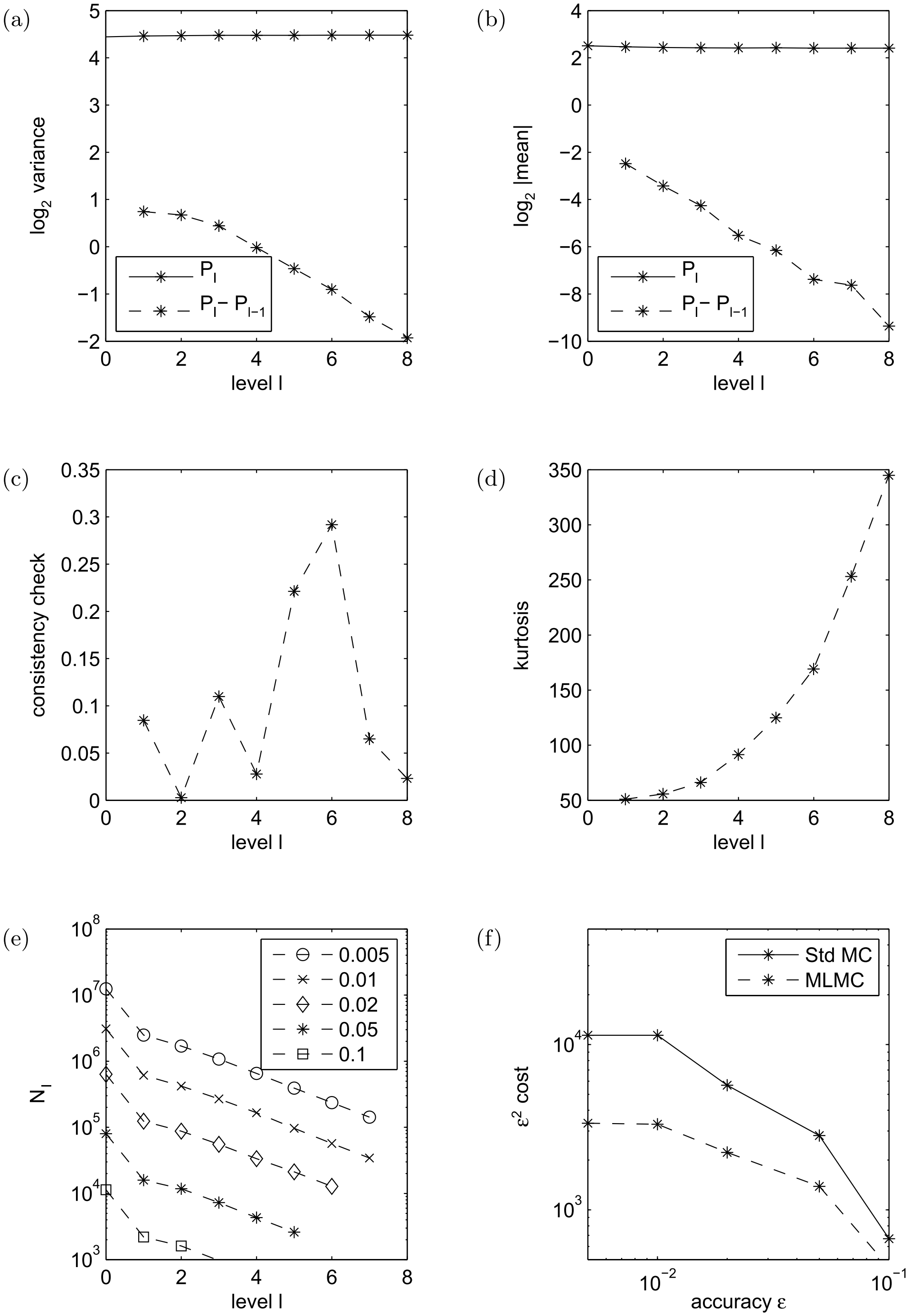

Figure 5.2. Numerical results for a digital call option using the Euler–Maruyama discretization of the GBM SDE.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

Table 5.1. Observed and theoretical convergence rates for the multilevel correction variance for scalar SDEs, using the Euler–Maruyama and Milstein discretizations. δ is any strictly positive constant.

	Euler–Maruyama		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	$O(h^{1/2})$	$O(h^{1/2}\log h)$	$O(h^{3/2})$	$o(h^{3/2-\delta})$

to the MLMC complexity being $O(\varepsilon^{-2.5})$. This slightly worse complexity is visible in plot (f), and the computational savings compared to the standard Monte Carlo method are lower.

Table 5.1 summarizes the observed variance convergence rate in numerical experiments for a number of different financial options; the Asian option is based on the average value of the underlying asset, the lookback is based on its maximum or minimum value, and the barrier is a discontinuous function of the maximum or minimum. The table also displays the theoretical results which have been obtained; the digital option analysis is due to Avikainen (2009) while the others are due to Giles, Higham and Mao (2009).

5.2. *Milstein discretization*

For Lipschitz payoffs, the variance V_ℓ for the natural multilevel estimator converges at twice the order of the strong convergence of the numerical approximation of the SDE. This immediately suggests that it would be better to replace the Euler–Maruyama discretization by the Milstein discretization (Giles 2008*a*) since it gives first-order strong convergence under certain conditions (see Theorem 10.3.5 in Kloeden and Platen 1992).

For a scalar SDE the Milstein discretization is

$$\widehat{S}_{n+1} = \widehat{S}_n + a(\widehat{S}_n, t_n)\, h + b(\widehat{S}_n, t_n)\, \Delta W_n + \tfrac{1}{2} b(\widehat{S}_n, t_n) \frac{\partial b}{\partial S}(\widehat{S}_n, t_n)(\Delta W_n^2 - h)$$

and Figure 5.3 demonstrates the improved results obtained for the call option based on a single underlying GBM asset. V_ℓ is now $O(h_\ell^2)$, leading to $\alpha = 1$, $\beta = 2$, $\gamma = 1$, and there is a significant reduction in the total computational cost compared to MLMC using the Euler–Maruyama discretization.

Because $\beta > \gamma$, the dominant computational cost is on the coarsest levels. Since these are low-dimensional, they are well suited to the use of quasi-Monte Carlo methods which are particularly effective in lower dimensions.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

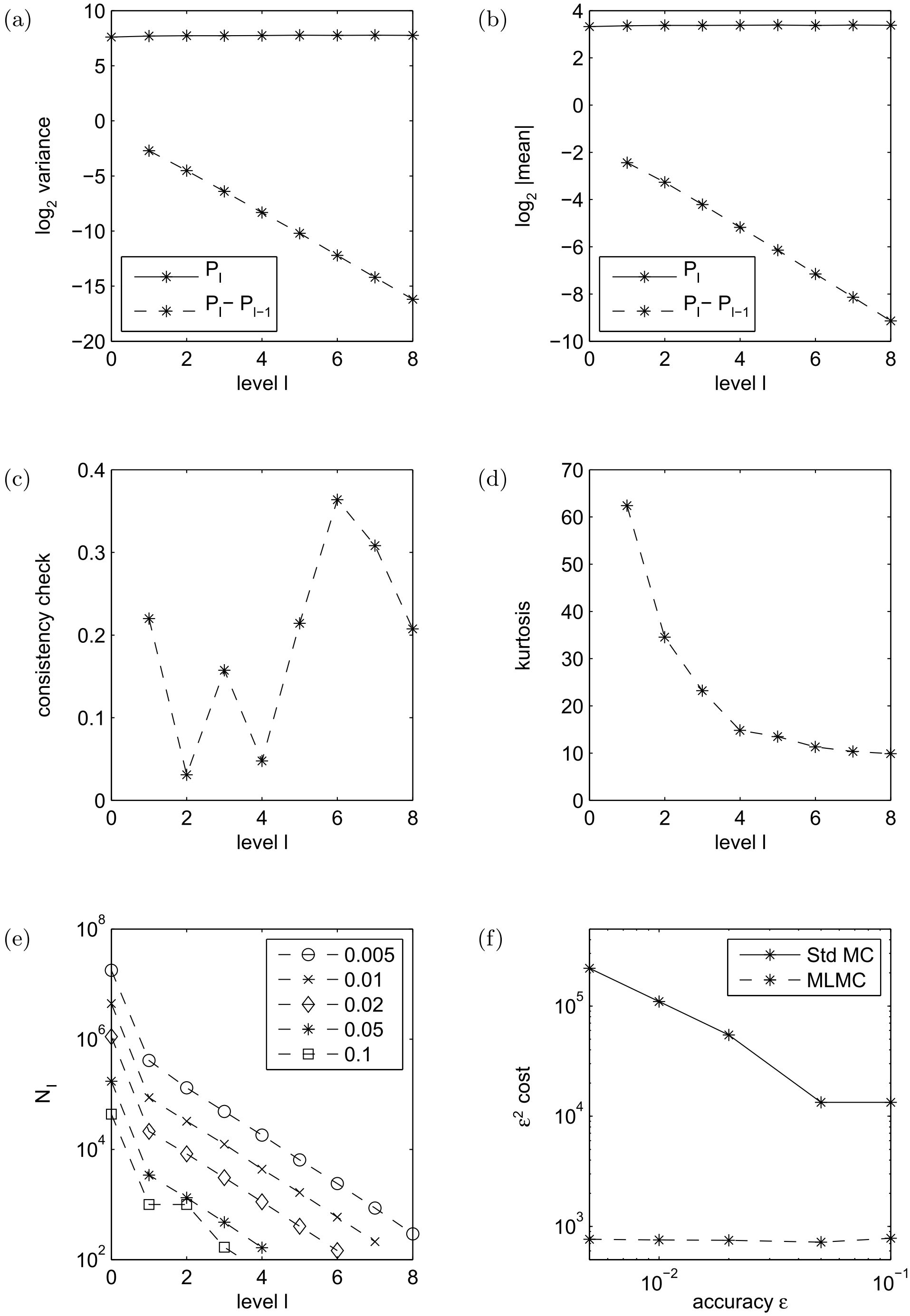

Figure 5.3. Numerical results for a European call option using the Milstein discretization of the GBM SDE.

This has been investigated by Giles and Waterhouse (2009) using a rank-1 lattice rule to generate the quasi-random numbers, randomization with 32 independent offsets to obtain confidence intervals, and a standard Brownian bridge construction of the increments of the driving Brownian process. The numerical results showed that MLMC on its own was better than QMC on its own, but the combination of the two was even better. The QMC treatment greatly reduced the variance per sample for the coarsest levels, resulting in significantly reduced costs overall. In the simplest case of a Lipschitz European payoff, the computational complexity was reduced from $O(\varepsilon^{-2})$ to approximately $O(\varepsilon^{-1.5})$.

Digital options

As with the Euler–Maruyama discretization, discontinuous payoffs pose a challenge to the multilevel Monte Carlo approach because small differences in the coarse and fine path simulations can lead to an $O(1)$ difference in the payoff function. In the case of a digital option, if we use the natural multilevel estimator then $P_\ell - P_{\ell-1} = O(1)$ for an $O(h_\ell)$ fraction of the paths, giving $V_\ell = O(h_\ell)$, and a kurtosis which is $O(h_\ell^{-1})$. To obtain a multilevel estimator with an improved variance convergence, we exploit the ideas of Section 2.1 and develop different estimators P^c and P^f for the coarse and fine paths, based on one of the following ideas:

- conditional expectation,
- splitting,
- change of measure.

The conditional expectation approach builds on a well-established technique for payoff smoothing which is used for pathwise sensitivity analysis (see, for example, pp. 399–400 in Glasserman 2004). We start by considering the fine path simulation, and make a slight change by using the Euler–Maruyama discretization for the final time-step, instead of the Milstein discretization. Conditional on the value $\widehat{S}_{N-1}$ which is the numerical approximation of S_{T-h} one time-step before the maturity T, the numerical approximation for the final value $\widehat{S}_N$ has a Gaussian distribution, and for a simple digital option the conditional expectation is known analytically. Thus the fine path payoff can be taken to be

$$P_\ell^f = \exp(-rT)\,\Phi\left(\frac{\widehat{S}^f_{N-1} + a(\widehat{S}^f_{N-1})h_\ell - K}{b(\widehat{S}^f_{N-1})\sqrt{h_\ell}}\right),$$

where Φ is the normal cumulative distribution function.

A similar treatment is used for the coarse path, except that in the final time-step, we re-use the known value of the Brownian increment for the second-last fine time-step, which corresponds to the first half of the final

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

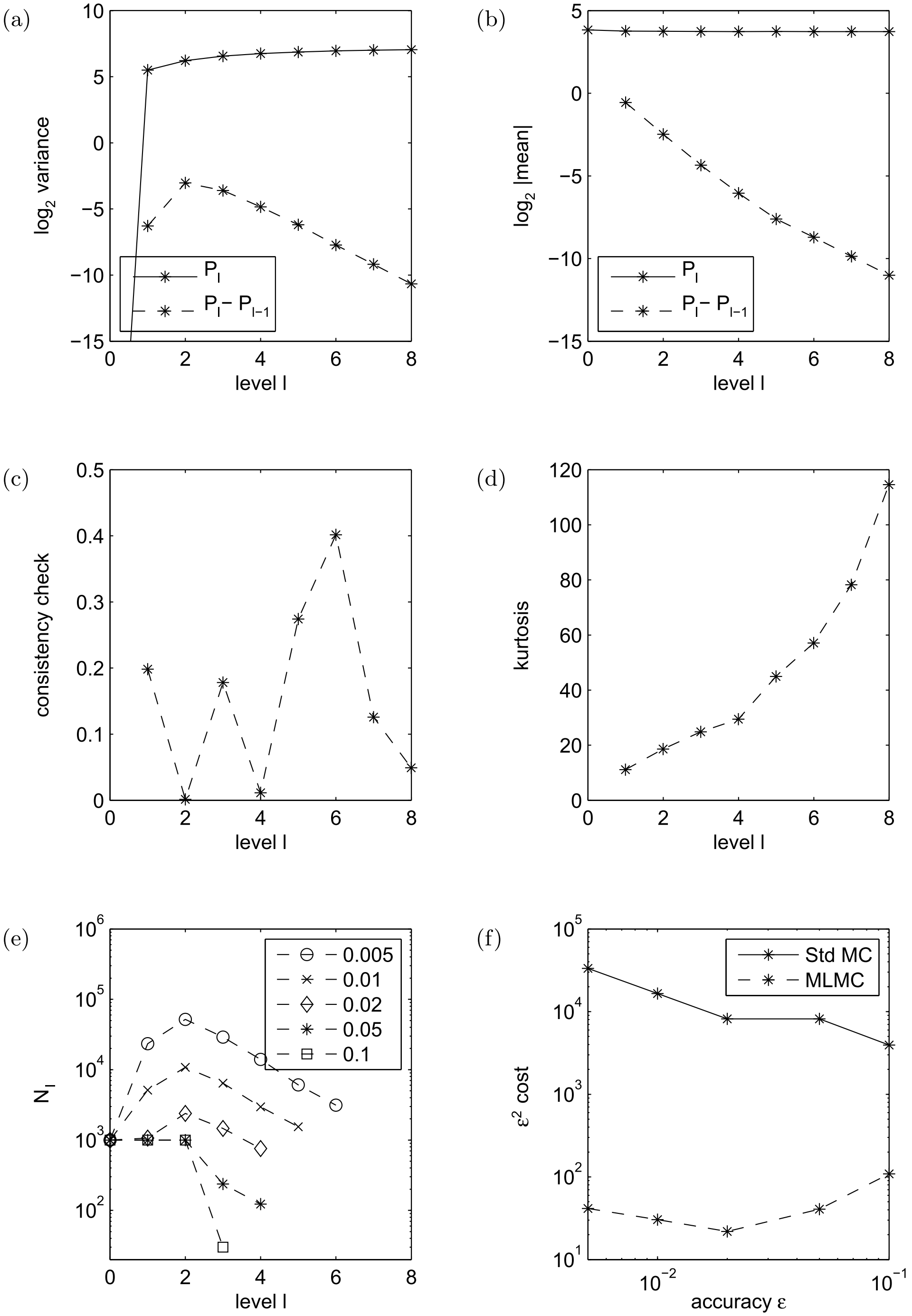

Figure 5.4. Numerical results for a digital call option using the Milstein discretization of the GBM SDE, and a conditional expectation MLMC estimator.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

coarse time-step. This results in the conditional distribution for the coarse path underlying at maturity matching that of the fine path to within $O(h)$, for both the mean and the standard deviation (Giles, Debrabant and Rößler 2013), and the corresponding coarse path payoff function is

$$P^c_{\ell-1} = \exp(-rT)\,\Phi\left(\frac{\widehat{S}^c_{N-2} + a(\widehat{S}^c_{N-2})h_{\ell-1} + b(\widehat{S}^c_{N-2})\sqrt{h_\ell} - K}{b(\widehat{S}^c_{N-2})\sqrt{h_\ell}}\right).$$

Consequently, the difference in payoff between the coarse and fine paths near the payoff discontinuity is $O(h^{1/2})$, giving a variance which is approximately $O(h^{3/2})$, and a kurtosis which is approximately $O(h^{-1/2})$. This leads to $\alpha = 1$, $\beta = 3/2$ and $\gamma = 1$. Since $\beta > \gamma$, the MLMC complexity is $O(\varepsilon^{-2})$.

This is all illustrated by the numerical results in Figure 5.4. One particularly interesting feature of these results is that there is zero variance on the coarsest level. This is because there is only one time-step on the coarsest level, and therefore the conditional expectation is taken immediately and every sample gives the same payoff.

It is very important in this conditional expectation formulation that $\mathbb{E}[P^c_{\ell-1}] = \mathbb{E}[P^f_{\ell-1}]$ so that the numerical payoff approximation on level $\ell-1$ has the same expected value regardless of whether it is the coarser or finer of the two levels. This ensures that the identity in equation (2.4) is respected so that the telescoping summation remains valid.

The conditional expectation technique works well in one dimension, where there is a known analytic value for the conditional expectation, but in multiple dimensions it may not be known. In this case, one can use the technique of 'splitting' (Asmussen and Glynn 2007). Here the conditional expectation is replaced by a numerical estimate, averaging over a number of sub-samples. That is, for each set of Brownian increments up to one fine time-step before the end, we use a number of samples of the final Brownian increment to produce an average payoff. If the number of sub-samples is chosen appropriately, the variance is the same, to leading order, without any increase in the computational cost, again to leading order. Because of its simplicity and generality, this is now the author's preferred approach. Furthermore, one can revert to using the Milstein approximation for the final time-step.

The change of measure approach is another approximation to the conditional expectation. Using the Euler–Maruyama approximation for the final time-step, the fine and coarse path conditional distributions at maturity are two very similar Gaussian distributions. Instead of following the splitting approach of taking corresponding samples from these two distributions, we can take a sample from a third Gaussian distribution (with a mean and variance perhaps equal to the average of the other two). This leads to the introduction of a Radon–Nikodym derivative for each path,

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

and the difference in the payoffs from the two paths is then due to the difference in their Radon–Nikodym derivatives.

In the specific context of digital options, this is a more complicated method to implement, and the resulting variance is no better. However, in other contexts a similar approach can be very effective.

Lookback and barrier options

Lookback and barrier options are financial options which depend on the minimum (or maximum) values of the underlying asset during the whole simulation interval $[0, T]$. A multilevel estimator based directly on the minimum (or maximum) of the values at the discrete time-steps will have a poor variance. This is because there is an $O(h^{1/2})$ variation in the asset value within each time-step of size h, and therefore there is an $O(h_\ell)$ difference on average between the minimum (or maximum) values for the coarse and fine paths. This results in an $O(h_\ell)$ variance for lookback options which are a Lipschitz function of the minimum (or maximum), and an even worse $O(h_\ell^{1/2})$ variance for barrier options which are a discontinuous function of the minimum (or maximum).

The key to achieving an improved variance is the definition of a Brownian bridge interpolant based on the approximation that the drift and volatility do not vary within the time-step (Giles 2008*a*). For the fine path, the interpolant for time-step $[t_n, t_n + h]$ is

$$\widehat{S}^f(t) = \widehat{S}^f_n + \lambda\big(\widehat{S}^f_{n+1} - \widehat{S}^f_n\big) + b^f_n\big(W(t) - W_n - \lambda\,(W_{n+1} - W_n)\big),$$

where $\lambda = (t - t_n)/h$ and $b^f_n \equiv b(\widehat{S}^f_n, t_n)$. This interpolant corresponds to a Brownian motion with constant drift and volatility, and for this there are standard results for the distribution of the minimum or maximum within each fine time-step (Glasserman 2004) which can be used to construct payoff approximations.

A similar construction can be used for the coarse path, but its time-step is twice as large, so the interpolant for the time interval $[t_n, t_n + 2h]$ is

$$\widehat{S}^c(t) = \widehat{S}^c_n + \lambda\big(\widehat{S}^c_{n+2} - \widehat{S}^c_n\big) + b^c_n\big(W(t) - W_n - \lambda\,(W_{n+2} - W_n)\big),$$

where $\lambda = (t - t_n)/(2h)$ and $b^c_n \equiv b(\widehat{S}^c_n, c_n)$. We can now evaluate this interpolant at time $t_n + h$, obtaining

$$\widehat{S}^c(t_{n+1}) = \tfrac{1}{2}\big(\widehat{S}^c_n + \widehat{S}^c_{n+2}\big) + \tfrac{1}{2}b^c_n\big((W_{n+1} - W_n) - (W_{n+2} - W_{n+1})\big),$$

using the Brownian increments $W_{n+1} - W_n$ and $W_{n+2} - W_{n+1}$ already generated for the fine path. The standard Brownian path results can then be used to obtain the coarse path payoff approximations.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

The full details are given in Giles (2008*a*); the outcome is that $\beta = 2$ for lookback options, and $\beta = 1.5$ for barrier options. Hence, we remain in the regime where $\beta > \gamma$ and the overall complexity is $O(\varepsilon^{-2})$.

Table 5.1 summarizes the convergence behaviour observed numerically for the different financial option types using the Milstein discretization, and the supporting numerical analysis by Giles *et al.* (2013).

5.3. Multi-dimensional SDEs

The discussion so far has been for scalar SDEs, but the computational benefits of Monte Carlo methods arise in higher dimensions. For multi-dimensional SDEs satisfying the usual commutativity condition (see, for example, p. 353 in Glasserman 2004) the Milstein discretization requires only Brownian increments for its implementation, and most of the analysis above carries over very naturally.

The only difficulties are with lookback and barrier options, where the classical results for the distribution of the minimum or maximum of a one-dimensional Brownian motion do not extend to the joint distribution of the minima or maxima of two correlated Brownian motions. However, if the financial option depends on the weighted average of a set of underlying assets, then the Brownian interpolation for each individual asset leads naturally to a Brownian interpolation for the average, and then the standard one-dimensional results can be used as usual.

Figures 5.5 and 5.6 show results for lookback and barrier options based on five underlying assets. Full details of the numerical construction are given in Giles (2009) and can also be viewed within the code Giles (2014). The MLMC variance is $O(h^2)$ for the lookback option, and $O(h^{3/2})$ for the barrier option.

With multi-dimensional SDEs which do not satisfy the commutativity condition, the Milstein discretization requires the simulation of Lévy areas, iterated Itô integrals which can only be relatively easily simulated in two dimensions. This is an unavoidable problem; the classical result of Clark and Cameron (1980) proves that $O(h^{1/2})$ strong convergence is the best that can be achieved in general using only Brownian increments.

Nevertheless, Giles and Szpruch (2014) have developed an antithetic treatment which achieves a very low variance despite the $O(h^{1/2})$ strong convergence. The estimator which is used is

$$Y_\ell = N_\ell^{-1} \sum_i \tfrac{1}{2}\big(P_\ell(\omega_i) + P_\ell(\omega_i^a)\big) - P_{\ell-1}(\omega_i).$$

Here ω_i represents the driving Brownian path, and ω_i^a is an antithetic counterpart defined by a time-reversal of the Brownian path within each coarse time-step. This results in the Brownian increments for the antithetic fine path being swapped relative to the original path. Lengthy analysis proves

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

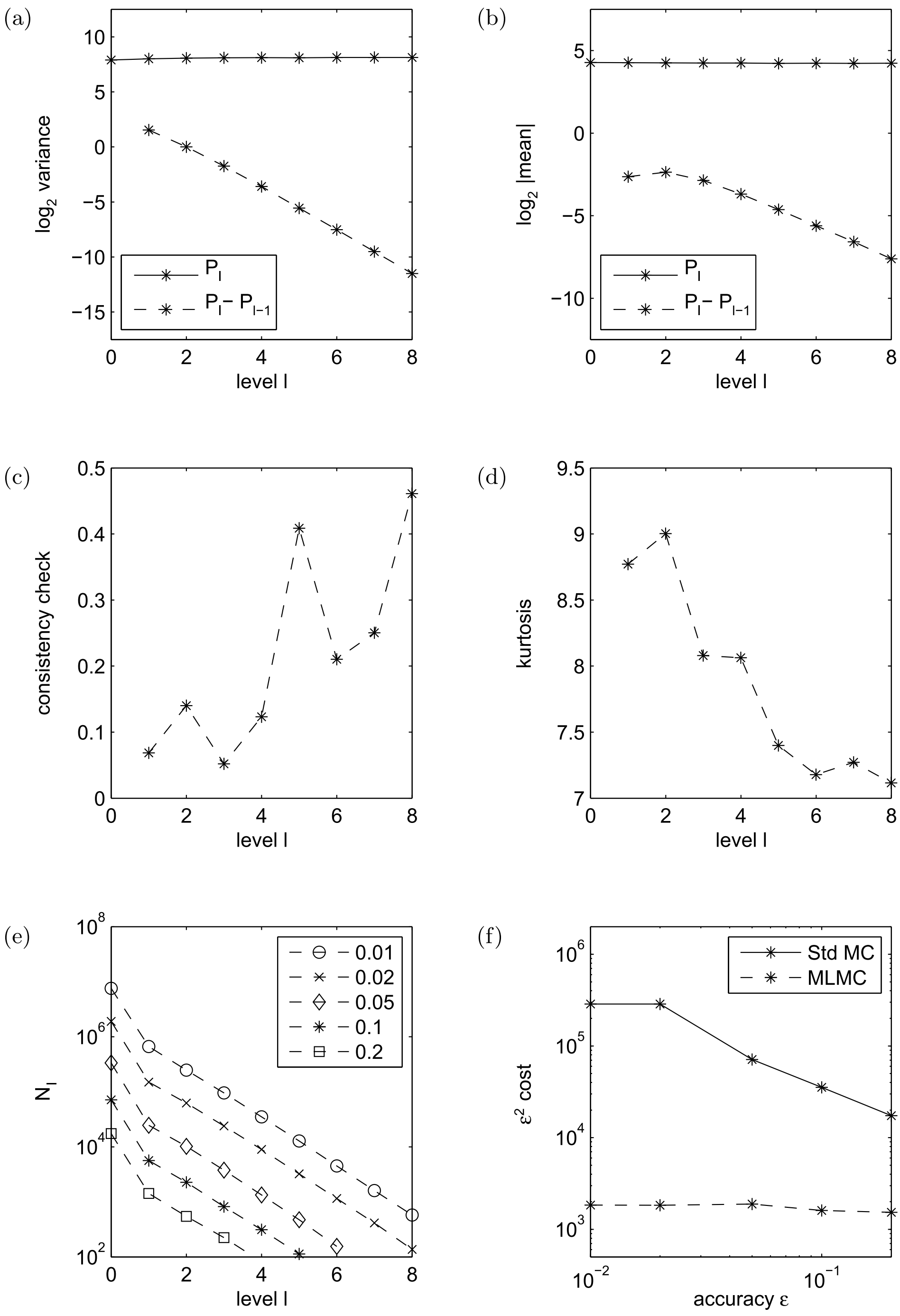

Figure 5.5. Numerical results for a lookback option on a basket of five GBM assets.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

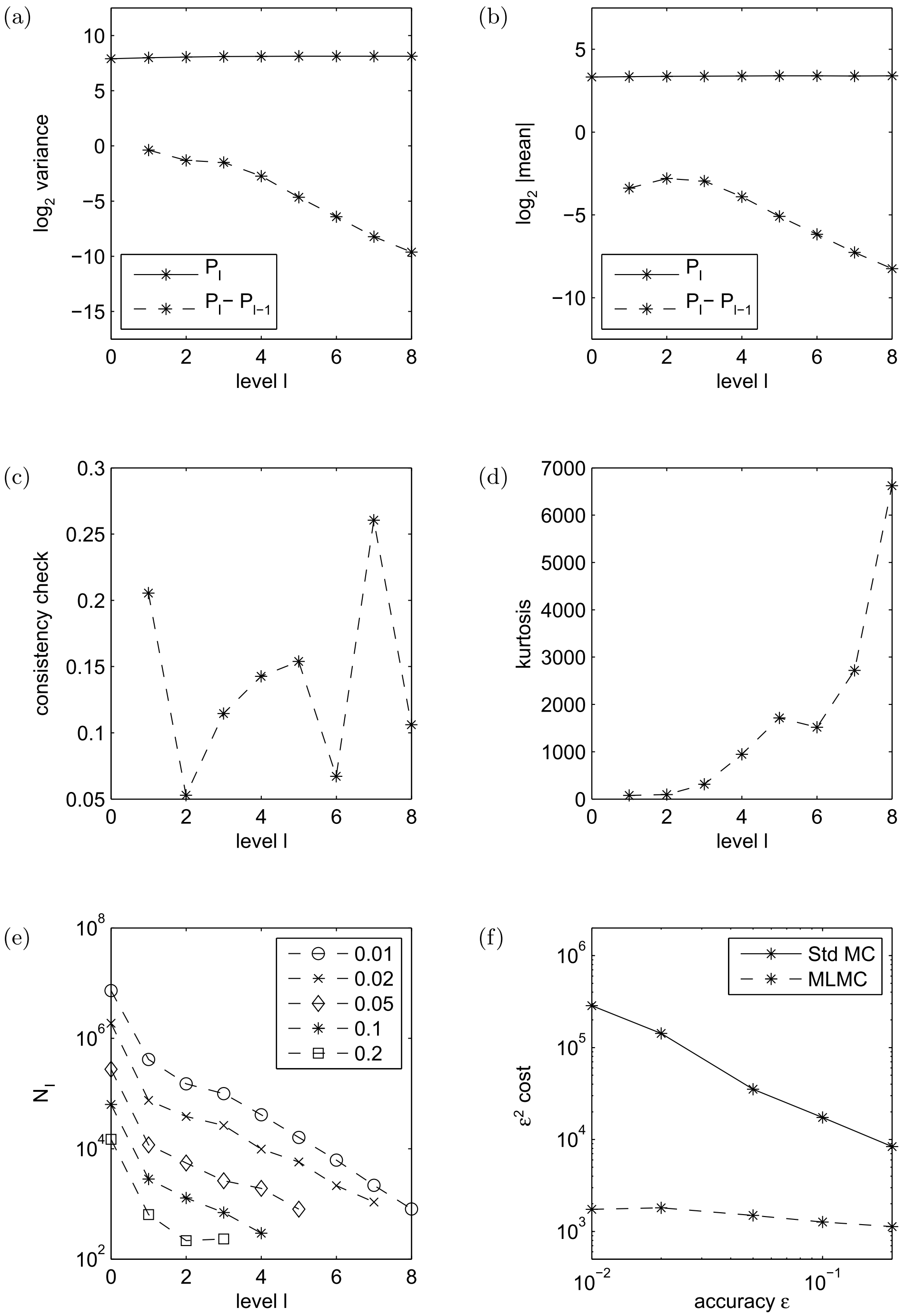

Figure 5.6. Numerical results for a barrier option on a basket of five GBM assets.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

that the average of the fine and antithetic paths is within $O(h)$ of the coarse path, and hence the multilevel variance is $O(h^2)$ for smooth payoffs, and $O(h^{3/2})$ for the standard European call option.

This treatment has been extended to handle lookback and barrier options (Giles and Szpruch 2013). This combines sub-sampling of the Brownian path to approximate the Lévy areas with sufficient accuracy to achieve $O(h^{3/4})$ strong convergence, with an antithetic treatment at the finest level of resolution to ensure that the average of the fine paths is within $O(h)$ of the coarse path.

5.4. Computing sensitivities

In computational finance, it is important to be able to compute sensitivities in addition to option prices. When the option payoff is continuous, the standard Monte Carlo approach is pathwise sensitivity analysis (Broadie and Glasserman 1996) (developed earlier as Infinitesimal Perturbation Analysis: see L'Ecuyer 1990), which considers linearized perturbations to the underlying path evolution, and the consequential perturbation to the payoff.

From a multilevel Monte Carlo point of view, the difficulty that this introduces is that the derivative of a call option payoff function is discontinuous. Hence, computing first-order sensitivities for a call option has similar difficulties to computing the option price for a digital option. Sensitivities for digital options can be obtained by first using the conditional expectation approach described previously, and then applying pathwise sensitivity analysis to this.

Full details on how to formulate appropriate MLMC estimators are given by Burgos and Giles (2012), and the numerical analysis of the resulting variance is given by Burgos (2014).

5.5. Exit times and Feynman–Kac formula

The simplest one-dimensional form of the Feynman–Kac formula states that if $u(x,t)$ is the solution of the parabolic PDE

$$\frac{\partial u}{\partial t} + a(x,t)\frac{\partial u}{\partial x} + \tfrac{1}{2}b^2(x,t)\frac{\partial^2 u}{\partial x^2} = 0,$$

for $x \in V$, subject to the terminal condition $u(x,T) = f(x)$, and the boundary condition $u(x,t) = g(x)$ for $x \in \partial V$, then $u(x,t)$ can also be written as the conditional expectation

$$u(x,t) = \mathbb{E}\big[f(S_T)\mathbf{1}_{\tau \geq T} + g(S_\tau)\mathbf{1}_{\tau < T} \mid S_t = x\big],$$

where S_t satisfies the SDE

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

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

and τ is the exit time

$$\tau = \inf_t \{t : S_t \notin V\}.$$

Because of this formula and its multi-dimensional generalization, and also because of other applications which are concerned with the time at which particles exit a particular domain, there is considerable interest in the estimation of exit times.

Higham *et al.* (2013) consider the estimation of $\mathbb{E}[\tau]$ using a multilevel approximation based on an Euler–Maruyama discretization. They prove that this can be achieved for very general multi-dimensional SDEs at a computational cost which is $O(\varepsilon^{-3}|\log \varepsilon|^{1/2})$. This is better than the $O(\varepsilon^{-4})$ complexity of a standard Monte Carlo simulation using the same numerical approximation of the exit time, but is not better than the complexity achieved by Gobet and Menozzi (2010) using a Monte Carlo simulation with a boundary correction which improves the weak order of convergence to first order.

Earlier research by Primozic (2011) demonstrates an $O(\varepsilon^{-2})$ complexity for a one-dimensional problem. This builds on the Milstein approximation for barrier options developed by Giles (2008*a*), using the approximate Brownian interpolation to determine the approximate probability that the underlying path crosses the boundary within each time-step. This improves both the weak convergence and the multilevel variance, giving $\alpha = 1$, $\beta = 3/2$ and $\gamma = 1$. This approach is capable of being extended to multi-dimensional SDEs which satisfy the commutativity condition which allows the use of the Milstein discretization without requiring the simulation of Lévy areas.

5.6. Stiff and highly nonlinear SDEs

In the numerical examples presented above, it is possible to use only one time-step on the coarsest level of approximation, and still obtain a sufficiently accurate path simulation so that the resulting payoff approximation is a useful control variate for the next level of approximation.

However, in other applications this may not be true. If the drift term has a characteristic time-scale of τ, as in a mean-reverting drift $(\theta - S_t)/\tau$, then when using the explicit Euler–Maruyama discretization the time-step h_0 on the coarsest level cannot be much larger than τ without encountering severe numerical stability problems. This is a natural restriction shared by explicit discretizations of ordinary differential equations.

A related problem is addressed by Hutzenthaler, Jentzen and Kloeden (2013), who are concerned with SDEs such as

$$\mathrm{d}S_t = -S_t^3 \,\mathrm{d}t + \mathrm{d}W_t,$$

which have a super-linear growth in the drift and/or the volatility. This again leads to numerical instability if a uniform time-step is used. Their

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

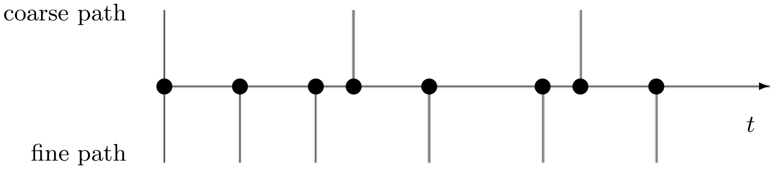

Figure 5.7. Generation of Brownian increments for a multilevel simulation with adaptive time-stepping.

solution is to introduce a slight modification to the Euler–Maruyama discretization which limits the size of the drift term on each level of approximation when S_t is large, to avoid this instability.

Other approaches to these problems include the use of drift-implicit methods (Dereich, Neuenkirch and Szpruch 2012, Higham, Mao and Stuart 2002), or a change or variables equivalent to the use of an integrating factor in ODEs (see the Heston treatment in Giles (2008*a*) which was suggested by Mark Broadie).

Another important approach, when the difficulties are only for extreme value of S_t, or for a short time interval, is to use adaptive time-stepping. Hoel, von Schwerin, Szepessy and Tempone (2012, 2014) construct a multilevel adaptive time-stepping discretization in which, for each stochastic instance ω, the time-steps used on level ℓ are a subdivision of those used on level $\ell-1$, which in turn are a subdivision of those on level $\ell-2$, and so on. By doing this, the payoff P_ℓ on level ℓ is the same regardless of whether one is computing $P_\ell - P_{\ell-1}$ or $P_{\ell+1} - P_\ell$, and therefore the MLMC telescoping summation is respected. Another notable aspect of their work is the use of adjoint/dual sensitivities to determine the optimal time-step size.

New unpublished work by Giles, Süli, Whittle and Ye implements adaptive time-stepping slightly differently. Instead of the time-steps on one level being a subdivision of those on the level above, it uses a completely independent adaptation on each level of refinement, with an adaptive time-step of the form

$$h_\ell = 2^{-\ell} H(\widehat{S}_n),$$

where $H(S)$ is independent of level. This results in time-steps which are not naturally nested. It may appear that this would cause difficulties in the MLMC implementation, but Figure 5.7 tries to illustrate that it does not. The underlying Brownian path needs to be sampled at a set of times which are the union of the simulation times used by the coarse and fine path. The independent Brownian increments can be simulated for each time interval, and summed to give $W(t)$ at the required times. An outline algorithm to implement this is given in Algorithm 3.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

Algorithm 3 Outline of a multilevel algorithm with adaptive time-stepping for the time interval $[0, T]$

```
$t := 0$
$t^c := 0$
$t^f := 0$
$\Delta W^c := 0$
$\Delta W^f := 0$

while $(t < T)$ do
  $t_{\text{old}} := t$
  $t := \min(t^c, t^f)$
  $h := t - t_{\text{old}}$
  $\Delta W := \sqrt{h}\, Z$
  $\Delta W^c := \Delta W^c + \Delta W$
  $\Delta W^f := \Delta W^f + \Delta W$

  if $t = t^c$ then
    update coarse path using $\Delta W^c$
    compute adapted coarse path time-step $h^c$
    $t^c := \min(t^c + h^c, T)$
    $\Delta W^c := 0$
  end if

  if $t = t^f$ then
    update fine path using $\Delta W^f$
    compute adapted fine path time-step $h^f$
    $t^f := \min(t^f + h^f, T)$
    $\Delta W^f := 0$
  end if

end while

compute $P_\ell - P_{\ell-1}$
```

5.7. CDF and density estimation

In some applications the objective is an estimation of the cumulative distribution function (CDF) or density of an output quantity P, rather than its expected value.

Kebaier and Kohatsu-Higa (2008) addressed the problem of constructing pointwise estimates of the density of the SDE solution S_T, using Malliavin calculus and a two-level variance reduction based on Kebaier's earlier work (Kebaier 2005).

Motivated by an interesting engineering application (Iliev, Nagapetyan and Ritter 2013), a multilevel approach to CDF and density estimation for

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

arbitrary output functions has been developed by Giles *et al.* (2015*b*). The CDF of a scalar output P is represented approximately by a polynomial spline function. Pointwise values of the CDF which are used as knot values in the spline construction correspond to expected values of an indicator function,

$$C(x) \equiv \mathbb{E}[\mathbf{1}_{P<x}] \equiv \mathbb{E}[H(x-P)],$$

where $H(x)$ is the Heaviside step function. To improve the multilevel variance, this is smoothed to give

$$C_\delta(x) = \mathbb{E}[(\,g((x-P)/\delta)],$$

where $g(x)$ is a continuous function with $g(x)=0$ for $x<-1$, and $g(x)=1$ for $x>1$. The parameter δ controls the degree of smoothing. As $\delta \to 0$, $g(x/\delta) \to H(x)$, and the accuracy improves but the variance of the multilevel estimator increases. Hence, this application has both spline approximation and smoothing errors in addition to the usual discretization and sampling errors. Balancing these errors gives the best accuracy for a given computational cost. The MLMC approach gives a substantial improvement in the order of complexity, and numerical examples demonstrate the savings which can be achieved.

The same paper also discusses the extension to density estimation. In this case, the density $\rho(x)$ of the scalar output P is given by

$$\rho(x) = \lim_{\delta\to 0} \mathbb{E}\big[\delta^{-1} g((x-P)/\delta)\big],$$

where $g(x)$ is a continuous function with $g(x)=0$ for $|x|>1$, and

$$\int_{-1}^{1} g(x)\;\mathrm{d}x = 1.$$

This has similarities with kernel density estimation (Silverman 1986), and can also be generalized to multi-dimensional outputs.

6. Jump diffusion and Lévy processes

6.1. Jump-diffusion processes

With finite activity jump-diffusion processes, such as in the Merton model (Merton 1976), it is natural to simulate each individual jump using a jump-adapted discretization (Mikulevicius and Platen 1988, Platen and Bruti-Liberati 2010) in which the Brownian diffusion between each jump is approximated using an Euler–Maruyama or Milstein approximation, and then each jump is simulated exactly.

If the jump activity rate is constant, then for each stochastic sample ω the jumps on the coarse and fine paths will occur at the same time, and

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

therefore the extension of the multilevel method is straightforward with the coarse and fine paths using the same underlying Brownian paths, and the same random variables to determine the jump times and strengths (Xia and Giles 2012).

If the jump rate is path-dependent then the situation is trickier. If there is a known upper bound to the jump rate, then one can use the 'thinning' approach of Glasserman and Merener (2004) in which a set of candidate jump times is simulated based on the constant upper bound, and then a subset of these are selected to be real jumps. The problem with the multilevel extension of this is that some candidate jumps will be selected for the coarse path but not for the fine path, or *vice versa*, leading to an $O(1)$ difference in the paths and hence the payoffs. Xia and Giles (2012) avoid this by using a change of measure to ensure that the jump times are the same for both paths; this introduces a Radon–Nikodym derivative into the payoff evaluation, but significantly reduces the multilevel correction variance.

6.2. More general processes

With infinite activity Lévy processes it is impossible to simulate each jump. One approach is to simulate the large jumps and either neglect the small jumps or approximate their effect by adding a Brownian diffusion term (Dereich 2011, Dereich and Heidenreich 2011, Marxen 2010). Following this approach, the cutoff δ_ℓ for the jumps which are simulated varies with level, and $\delta_\ell \to 0$ as $\ell \to \infty$ to ensure that the bias converges to zero. In the multilevel treatment, when simulating $P_\ell - P_{\ell-1}$ the jumps fall into three categories. The ones which are larger than $\delta_{\ell-1}$ get simulated in both the fine and coarse paths. The ones which are smaller than δ_ℓ are either neglected for both paths or approximated by the same Brownian increment. The difficulty is in the intermediate range $[\delta_\ell, \delta_{\ell-1}]$, in which the jumps are simulated for the fine path but neglected or approximated for the coarse path. This is what leads to the difference in path simulations, and contributes to a non-zero value for $P_\ell - P_{\ell-1}$.

Alternatively, for many SDEs driven by a Lévy process it is possible to directly simulate the increments of the Lévy process over a set of uniform time-steps (Schoutens 2003), in exactly the same way as one simulates Brownian increments. When estimating the price of European options which depend only on the final value of the Lévy process, there is no need for MLMC as one can directly simulate the final value. However, multilevel is still very useful for path-dependent financial options such as Asian, lookback and barrier options, and there has been good progress in the numerical analysis of these applications (Xia 2014, Xia and Giles 2014). Table 6.1 gives the convergence rates obtained from the numerical analysis; numerical experiments support these results but suggest they may not be sharp in some cases.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

Table 6.1. Numerical analysis convergence rates for the multilevel variance $V_\ell \equiv \mathbb{V}[P_\ell - P_{\ell-1}]$ for variance-gamma, NIG, and spectrally negative α-stable processes and different financial options. δ is any strictly positive constant.

	VG	NIG	α-stable
Asian	$O(h^2)$	$O(h^2)$	$O(h^2)$
lookback	$O(h)$	$O(h\lvert\log h\rvert)$	$O(h^{2/\alpha-\delta})$
barrier	$o(h^{1-\delta})$	$o(h^{1/2-\delta})$	$o(h^{1/\alpha-\delta})$

For other Lévy processes, it may be possible in the future to simulate the increments by constructing near-perfect (*i.e.*, to within the limits of finite precision computer arithmetic) approximations to the inverse of the cumulative distribution function. Where this is possible, it may be the best approach to achieve a perfect coupling between the coarse and fine path simulations, since the increments of the driving Lévy process for the coarse path can be obtained trivially by summing the increments for the fine path.

Finally, we mention the work of Ferreiro-Castilla, Kyprianou, Scheichl and Suryanarayana (2014), who develop a multilevel approach for barrier options using a Wiener–Hopf factorization technique to sample exactly from the joint distribution of the terminal value of a Lévy process at an exponential random stopping time, and its maximum (or minimum) over the intervening interval. The particularly novel aspect here is the random terminal time; by using multiple shorter random periods it is possible, because of the Central Limit Theorem, to get closer to a desired fixed terminal time, but at an increased cost. In the multilevel coupling, the thinning approach of Glasserman and Merener (2004) is again used to link the coarse and fine path simulations. Numerical analysis for lookback options gives a complexity which is $O(\varepsilon^{-3})$ for processes of bounded variation, and $O(\varepsilon^{-4})$ for processes of unbounded variation. Numerical experiments for barrier options demonstrate a similar complexity.

7. PDEs and SPDEs

After the initial work with MLMC for SDEs, it was immediately clear that it was equally applicable to SPDEs, and indeed the computational savings would be greater because the cost of a single sample increases more rapidly with grid resolution for SPDEs with higher space–time dimension. The first published work was by Graubner (2008) on parabolic SPDEs; since then there has been a variety of papers on elliptic (Barth, Schwab and Zollinger

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

2011, Charrier, Scheichl and Teckentrup 2013, Cliffe *et al.* 2011, Teckentrup, Scheichl, Giles and Ullmann 2013), parabolic (Barth, Lang and Schwab 2013, Giles and Reisinger 2012), hyperbolic (Mishra, Schwab and Sukys 2012) SPDEs, as well as for mixed elliptic–hyperbolic systems (Efendiev, Iliev and Kronsbein 2013, Müller, Jenny and Meyer 2013).

In almost all of this work, the construction of the multilevel estimator is quite natural, using a geometric sequence of grids and the usual estimators for $P_\ell - P_{\ell-1}$. It is the numerical analysis of the variance of the multilevel estimator which is often very challenging.

7.1. Two simple examples

The MATLAB codes accompanying this article include two simple PDE examples.

The first is really a boundary-value ODE application, but it can be viewed as a one-dimensional elliptic PDE, with random coefficients and random forcing. The equation is

$$\frac{\mathrm{d}}{\mathrm{d}x}\left(c(x)\frac{\mathrm{d}u}{\mathrm{d}x}\right) = -50\,Z^2,$$

on $0 < x < 1$, with boundary data $u(0) = u(1) = 0$. Z is a normal random variable with zero mean and unit variance, and

$$c(x) = 1 + a\,x,$$

with a being a uniform random variable on the unit interval $(0,1)$. The output quantity of interest is chosen to be

$$P = \int_0^1 u(x)\;\mathrm{d}x.$$

The multilevel implementation is very easy. Level ℓ uses a uniform grid with spacing $h_\ell = 2^{-(\ell+1)}$, so there is just one interior grid point on the coarsest level. A simple second-order central difference approximation is used (equivalent to a finite element approximation with a 1-point quadrature).

The uniform second-order accuracy means that there is a constant K such that $|P - P_\ell| < K\,h_\ell^2$, and therefore we have $\alpha = 2$, $\beta = 4$ and $\gamma = 1$, resulting in an $O(\varepsilon^{-2})$ complexity. All of these features can be verified in the numerical results in Figure 7.1.

The second example is a one-dimensional parabolic SPDE, in which $u(x,t)$ satisfies the SPDE

$$\mathrm{d}u = \frac{\partial^2 u}{\partial x^2}\;\mathrm{d}t + 10\;\mathrm{d}W,$$

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

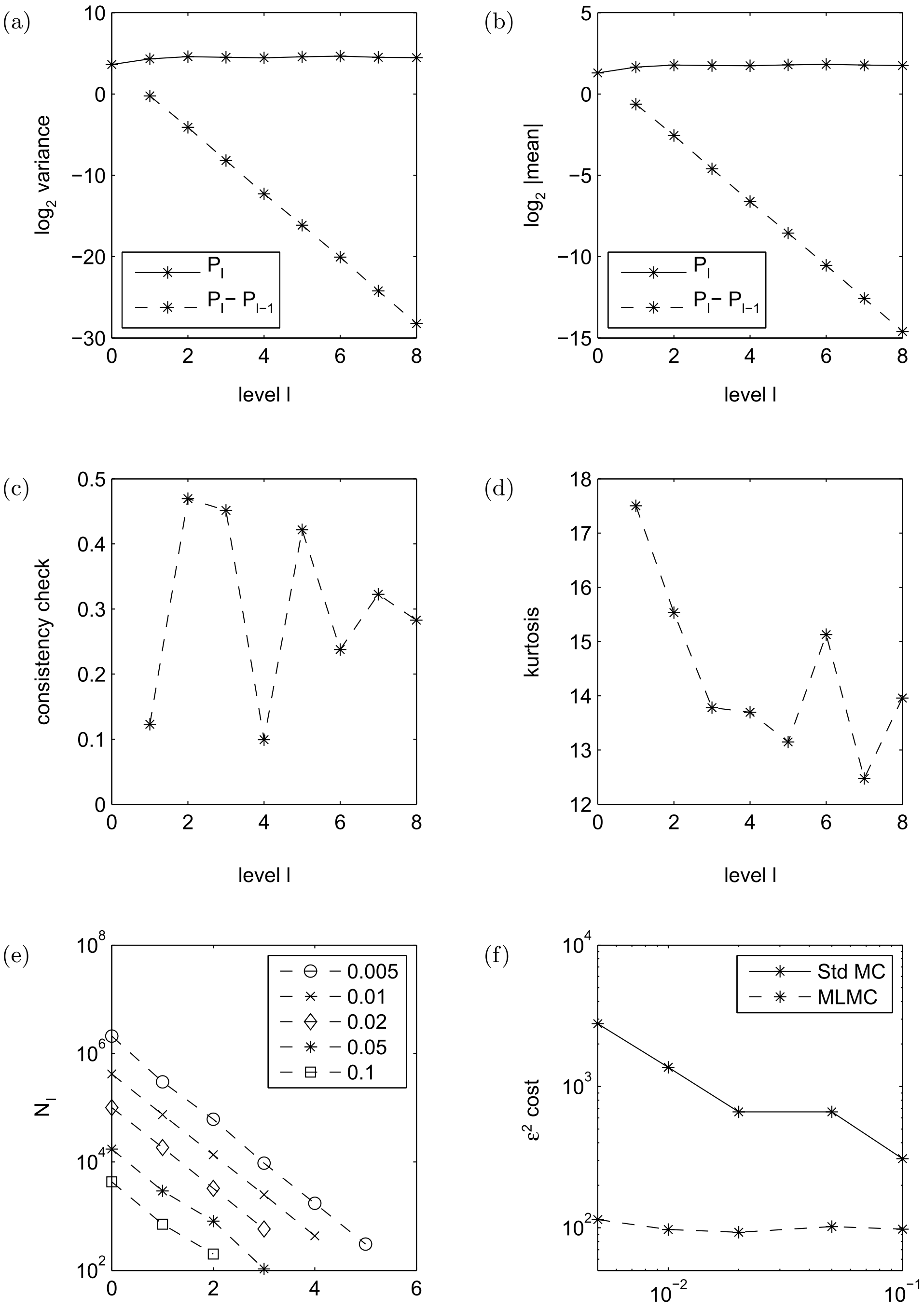

Figure 7.1. Numerical results for a one-dimensional elliptic PDE with random coefficients and random forcing.

on the domain $0 < x < 1$ with boundary data $u(0,t) = u(1,t) = 0$ and initial data $u(x,0) = 0$. The output functional is chosen to be

$$P = \int_0^1 u^2(x, 0.25).$$

The multilevel implementation is again very easy. An Euler–Maruyama time discretization with time-step k, combined with a second-order space discretization with uniform grid spacing h, gives the discrete approximation

$$u_j^{n+1} = u_j^n + \frac{k}{h^2}\left(u_{j+1}^n - 2u_j^n + u_{j-1}^n\right) + 10\,\Delta W^n.$$

The level ℓ approximation uses

$$h_\ell = 2^{-(\ell+1)}, \quad k_\ell = h_\ell^2/4.$$

Keeping $k_\ell/h_\ell^2 = 1/4$ ensures the explicit numerical discretization is stable on all levels. Since the number of grid points doubles on each level, and the number of time-steps increases by factor 4, the cost per sample increases by factor 8, giving $\gamma = 3$.

Because the stochastic forcing is additive, that is, the volatility does not depend on $u(x,t)$, the Euler–Maruyama discretization is actually equivalent to a Milstein discretization and hence the time discretization errors are $O(k)$. This is of the same order as the spatial discretization errors which are $O(h^2)$, and therefore the solution error is $O(2^{-2\ell})$ and hence $\alpha = 2$ and $\beta = 4$. This leads to the optimal complexity of $O(\varepsilon^{-2})$.

These features are again confirmed in the numerical results shown in Figure 7.2.

7.2. Elliptic SPDE

The largest amount of research on multilevel for SPDEs has been for elliptic PDEs with random coefficients. The PDE typically has the form

$$-\nabla\cdot\left(\kappa(\mathbf{x},\omega)\nabla p(\mathbf{x},\omega)\right) = 0, \quad \mathbf{x}\in D.$$

with Dirichlet or Neumann boundary conditions on the boundary ∂D. For subsurface flow problems, such as the modelling of groundwater flow in nuclear waste repositories, the diffusivity (or permeability) κ is often modelled as a log-normal random field, that is, $\log\kappa$ is a Gaussian field with a uniform mean (which we will take to be zero for simplicity) and a covariance function of the general form $R(\mathbf{x},\mathbf{y}) = r(\mathbf{x}-\mathbf{y})$. Samples of $\log k$ are provided by a Karhunen–Loève expansion,

$$\log\kappa(\mathbf{x},\omega) = \sum_{n=0}^{\infty}\sqrt{\theta_n}\;\xi_n(\omega)\;f_n(\mathbf{x}),$$

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

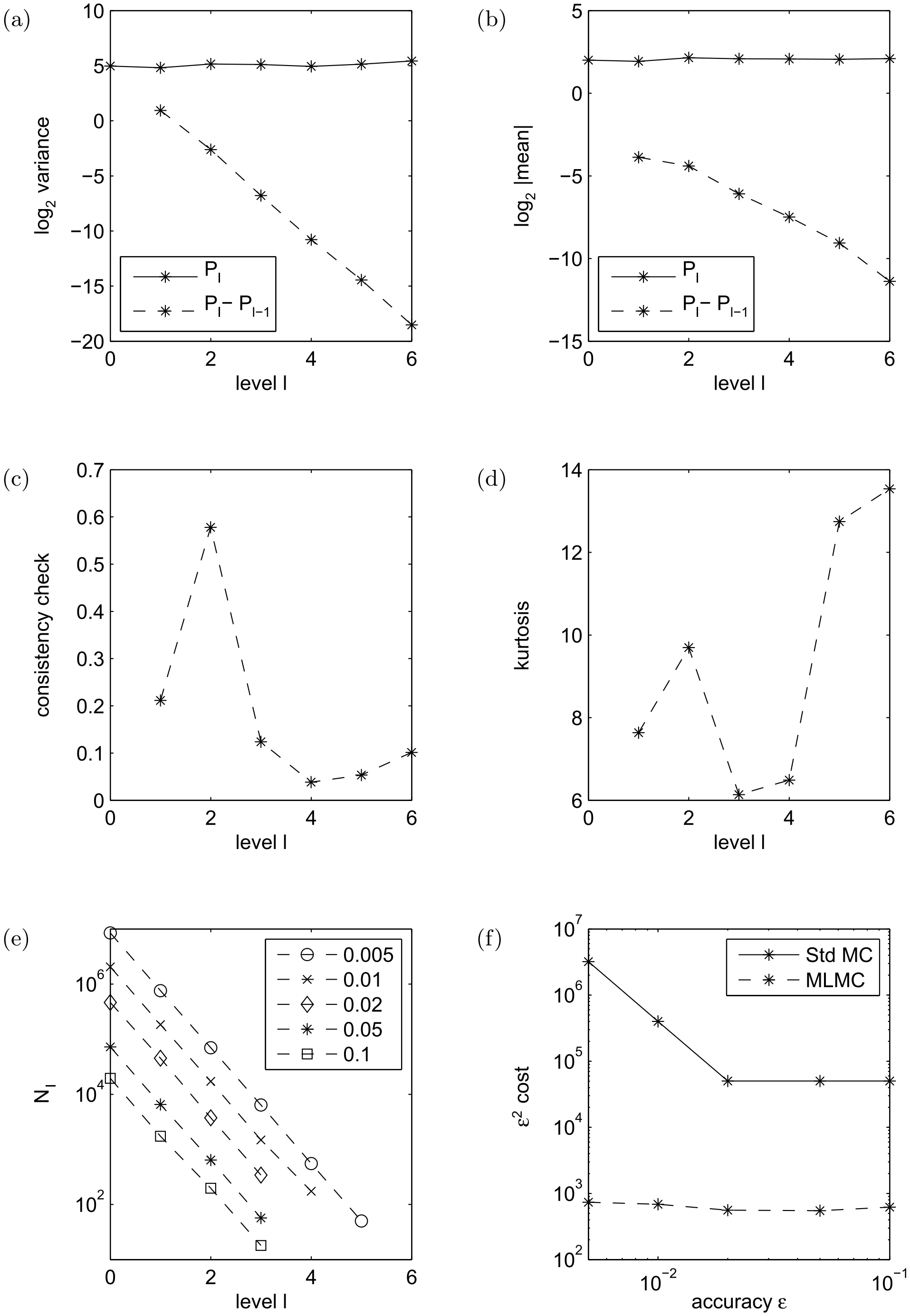

Figure 7.2. Numerical results for a one-dimensional parabolic SPDE with Brownian forcing.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

where θ_n are the eigenvalues of $R(\mathbf{x},\mathbf{y})$ in decreasing order, f_n are the corresponding eigenfunctions, and ξ_n are independent unit normal random variables. However, it is more efficient to generate them using a circulant embedding technique which enables the use of FFTs (Dietrich and Newsam 1997).

The multilevel treatment is straightforward. The spatial grid resolution is doubled on each level. Using the Karhunen–Loève generation, the expansion is truncated after K_ℓ terms, with K_ℓ increasing with level (Teckentrup *et al.* 2013); a similar approach has also been used with the circulant embedding generation.

In both cases, $\log \kappa$ is generated using a row-vector of independent unit normal random variables ξ. The variables for the fine level can be partitioned into those for the coarse level $\xi_{\ell-1}$, plus some additional variables z_ℓ, giving $\xi_\ell = (\xi_{\ell-1}, z_\ell)$.

The numerical analysis of the multilevel approach for these elliptic SPDE applications is challenging because the diffusivity is unbounded, but Charrier *et al.* (2013) have successfully analysed it for certain output functionals, and Teckentrup *et al.* have further developed the analysis for other output functionals and more general log-normal diffusivity fields (Teckentrup *et al.* 2013, Teckentrup 2013).

7.3. Parabolic SPDE

Giles and Reisinger (2012) consider an unusual SPDE from credit default modelling,

$$\mathrm{d}p = -\mu \,\frac{\partial p}{\partial x}\,\mathrm{d}t + \tfrac{1}{2}\frac{\partial^2 p}{\partial x^2}\,\mathrm{d}t - \sqrt{\rho}\,\frac{\partial p}{\partial x}\,\mathrm{d}M_t, \quad x>0$$

subject to boundary condition $p(0,t)=0$. Here $p(x,t)$ represents the probability density function for firms being a distance x from default at time t. The diffusive term is due to idiosyncratic factors affecting individual firms, while the stochastic term due to the scalar Brownian motion M_t corresponds to the systemic movement due to random market effects on all firms. The payoff corresponds to different tranches of a credit derivative which depends on the integral $\int_0^\infty p(x,t)\,\mathrm{d}x$ at a set of discrete times.

A Milstein time discretization with time-step k, and a central space discretization of the spatial derivatives with uniform spacing h give the numerical approximation

$$\begin{aligned} p_j^{n+1} = p_j^n &- \frac{\mu\, k + \sqrt{\rho\, k}\, Z_n}{2h}\left(p_{j+1}^n - p_{j-1}^n\right) \\ &+ \frac{(1-\rho)\,k + \rho\, k\, Z_n^2}{2h^2}\left(p_{j+1}^n - 2p_j^n + p_{j-1}^n\right), \end{aligned}$$

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

where $p_j^n \approx p(j\,h, n\,k)$, and the Z_n are standard normal random variables so that $\sqrt{h}\; Z_n$ corresponds to an increment of the driving scalar Brownian motion.

The multilevel implementation is very straightforward. As with the model parabolic example discussed previously, $k_\ell = k_{\ell-1}/4$ and $h_\ell = h_{\ell-1}/2$ due to numerical stability considerations which are analysed in the paper. The coupling between the coarse and fine samples comes from summing the fine path Brownian increments to give the increments for the coarse path, exactly the same as for SDEs The computational cost increases by factor 8 on each level, and numerical experiments indicate that the variance decreases by factor 8, so the overall computational complexity to achieve an $O(\varepsilon)$ RMS error is again $O(\varepsilon^{-2}(\log\varepsilon)^2)$.

This work has been extended by Bujok and Reisinger (2012) to include a systemic jump-diffusion term. They also switch from continuous monitoring of default, which gives the boundary condition $p(0,t) = 0$ for all t, to discrete monitoring which introduces the boundary condition $p(x,t) = 0$ for $x \leq 0$ for each discrete default date t.

7.4. Reduced basis approximation

New research by Vidal-Codina *et al.* (2014) uses a very different approach to SPDE approximation. They start with a high-accuracy finite element approximation

$$A(\omega)\,\mathbf{u}(\omega) = \mathbf{f}(\omega),$$

where ω represents the dependence on the stochastic input parameters. They then use a reduced basis approximation with a fixed set of basis functions $\mathbf{u}_k$

$$\mathbf{u}(\omega) \approx \sum_{k=1}^{K} u_k(\omega)\,\mathbf{u}_k,$$

which leads to a much smaller system of equations

$$A_K(\omega) \begin{pmatrix} u_1 \\ u_2 \\ \cdots \\ u_K \end{pmatrix} = \mathbf{f}_K(\omega).$$

The output of interest is a function of the solution, so that they are interested in estimating $\mathbb{E}[f(\mathbf{u})]$. There is an accuracy–cost tradeoff here: larger values for K lead to greater accuracy, but at a much larger cost. The MLMC treatment makes K a function of the level ℓ, so that most simulations are performed with a small value for K, and relatively few with a large value.

In this application, geometric MLMC is not appropriate and so they face the question of how to choose the levels. As discussed already in Section 2.6,

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

they start with a set of candidate levels and use a number of trial samples to obtain an estimate of the variances $V_{\ell_1,\ell_2} \equiv \mathbb{V}[P_{\ell_1} - P_{\ell_2}]$ for all pairs of levels. They then use this to determine the optimal subset of levels for the main computation.

8. Continuous-time Markov chains

Anderson and Higham (2012) developed a very interesting application of MLMC to continuous-time Markov chain simulation. Although they present their work in the context of stochastic chemical reactions, when species concentrations are extremely low and so stochastic effects become significant, they point out that the method has wide applicability in other areas.

When there is just one chemical reaction, the 'tau-leaping' method (which is essentially the Euler–Maruyama method, approximating the reaction rate as being constant throughout the time-step) gives the discrete equation

$$\mathbf{x}_{n+1} = \mathbf{x}_n + P(h\,\lambda(\mathbf{x}_n)),$$

where h is the time-step, $\lambda(\mathbf{x}_n)$ is the reaction rate (or propensity function), and $P(t)$ represents a unit-rate Poisson random variable over time interval $[0, t]$. If this equation defines the fine path in the multilevel simulation, then the coarse path, with double the time-step, is given by

$$\mathbf{x}^c_{n+2} = \mathbf{x}^c_n + P(2h\,\lambda(\mathbf{x}^c_n))$$

for even time-steps n.

The question then is how to couple the coarse and fine path simulations. The key observation in Anderson and Higham (2012), based on earlier work in a different context by Kurtz (1982), is that for any $t_1, t_2 > 0$, the sum of two independent Poisson variates $P(t_1)$, $P(t_2)$ is equivalent in distribution to $P(t_1+t_2)$. Based on this, the first step is to express the coarse path Poisson variate as the sum of two Poisson variates, $P(h\,\lambda(\mathbf{x}^c_n))$ corresponding to the first and second fine path time-steps. For the first of the two fine time-steps, the coarse and fine path Poisson variates are coupled by defining two Poisson variates based on the minimum of the two reaction rates, and the absolute difference,

$$P_1 = P\big(h\min(\lambda(\mathbf{x}_n), \lambda(\mathbf{x}^c_n))\big), \quad P_2 = P\big(h|\lambda(\mathbf{x}_n) - \lambda(\mathbf{x}^c_n)|\big),$$

and then using P_1 as the Poisson variate for the path with the smaller rate, and $P_1 + P_2$ for the path with the larger rate. This elegant approach naturally gives a small difference in the Poisson variates when the difference in rates is small, and leads to a very effective multilevel algorithm with a correction variance which is $O(h)$, leading to an $O(\varepsilon^{-2}(\log\varepsilon)^2)$ complexity.

In their paper, Anderson and Higham (2012) treat more general systems with multiple reactions. They also include an additional coupling at the finest level to the exact Stochastic Simulation Algorithm developed by Gillespie (1976), which updates the reaction rates after every single reaction. Hence, their overall multilevel estimator is unbiased, unlike the estimators discussed earlier for SDEs, and the complexity is reduced to $O(\varepsilon^{-2})$ because the number of levels remains fixed as $\varepsilon \to 0$. They give a complete numerical analysis of the variance of their multilevel algorithm; this has been further sharpened in more recent work (Anderson, Higham and Sun 2014). Because stochastic chemical simulations typically involve 1000s of reactions, the multilevel method is particularly effective in this context, providing computational savings in excess of a factor of 100 (Anderson and Higham 2012).

They also give an interesting numerical example in which an approximate model with fewer reactions/reactants is used as a control variate for the full system. This kind of multilevel modelling is another possibility which could be considered in a wide variety of circumstances.

Moraes, Tempone and Vilanova (2015) have extended these ideas by adaptively switching between tau-leaping which is suitable when reaction rates are high, and exact simulation which is more appropriate when reaction rates are low relative to the time-step. They also develop a more accurate estimator for the multilevel variance, which otherwise suffers due to high kurtosis on the finest grid levels, and a method for controlling the probability that the tau-leaping leads to $\mathbf{x}$ becoming negative.

In a second paper, Moraes, Tempone and Vilanova (2014) generalize the adaptive switching between tau-leaping and exact simulation to treat each reaction separately. This is very beneficial when different reactions have very different propensities. They also introduce a novel control variate for the coarsest simulation level which significantly reduces the total simulation cost. Numerical examples demonstrating impressive cost savings of up to 1000× relative to standard methods.

The non-nested adaptive time-stepping approach described in Section 5.6 for SDEs is equally applicable in this setting. As described in Giles, Lester and Whittle (2015*a*) and Lester, Yates, Giles and Baker (2015), the construction is exactly the same as illustrated in Figure 5.7, but with Poisson variates for each time interval instead of Brownian increments. This adaptive time-stepping can be very helpful in cases in which propensities vary greatly in time.

9. Nested simulation

9.1. MLMC treatment

In nested simulations we are interested in estimating quantities of the form

$$\mathbb{E}_Z\big[f\big(\mathbb{E}_W[g(Z,W)]\big)\big],$$

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

where Z is an outer random variable, and $\mathbb{E}_W[g(Z,W)]$ is a conditional expectation with respect to an independent inner random variable W.

For example, in some financial applications Z which represents different risk *scenarios*, $\mathbb{E}_W[g(Z,W)]$ represents exposure, conditional on the scenario Z, and f might correspond to the loss in excess of the capital reserves, so that $\mathbb{E}_Z\big[f\big(\mathbb{E}_W[g(Z,W)]\big)\big]$ is the expected shortfall.

This can be simulated using nested Monte Carlo simulation with N outer samples $Z^{(n)}$, M inner samples $W^{(m,n)}$ and a standard Monte Carlo estimator:

$$Y = N^{-1}\sum_{n=1}^{N} f\Bigg(M^{-1}\sum_{m=1}^{M} g\big(Z^{(n)},W^{(m,n)}\big)\Bigg).$$

Note that to improve the accuracy of the estimate we need to increase both M and N, and this will significantly increase the cost.

An MLMC implementation is straightforward; on level ℓ we can use $M_\ell = 2^\ell$ inner samples. To construct a low-variance estimate for $\mathbb{E}[P_\ell - P_{\ell-1}]$ where

$$\mathbb{E}[P_\ell] \equiv \mathbb{E}_Z\Bigg[f\Bigg(M_\ell^{-1}\sum_m g\big(Z,W^{(m)}\big)\Bigg)\Bigg],$$

we use an *antithetic* approach, and split the M_ℓ samples of W for the ‘fine’ value into two subsets of size $M_{\ell-1}$ for the ‘coarse’ value:

$$\begin{aligned} Y_\ell = N_\ell^{-1}\sum_{n=1}^{N_\ell} \Bigg\{ & f\Bigg(M_\ell^{-1}\sum_{m=1}^{M_\ell} g\big(Z^{(n)},W^{(m,n)}\big)\Bigg) \\ & -\tfrac{1}{2} f\Bigg(M_{\ell-1}^{-1}\sum_{m=1}^{M_{\ell-1}} g\big(Z^{(n)},W^{(m,n)}\big)\Bigg) \\ & -\tfrac{1}{2} f\Bigg(M_{\ell-1}^{-1}\sum_{m=M_{\ell-1}+1}^{M_\ell} g\big(Z^{(n)},W^{(m,n)}\big)\Bigg)\Bigg\}. \end{aligned}$$

Note that this has the correct expectation, that is, $\mathbb{E}[Y_\ell] = \mathbb{E}[P_\ell - P_{\ell-1}]$.

If we define

$$\begin{aligned} M_{\ell-1}^{-1}\sum_{m=1}^{M_{\ell-1}} g\big(Z^{(n)},W^{(m,n)}\big) &= \mathbb{E}\big[g\big(Z^{(n)},W\big)\big] + \Delta g_1^{(n)}, \\ M_{\ell-1}^{-1}\sum_{m=M_{\ell-1}+1}^{M_\ell} g\big(Z^{(n)},W^{(m,n)}\big) &= \mathbb{E}\big[g\big(Z^{(n)},W\big)\big] + \Delta g_2^{(n)}, \end{aligned}$$

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

then if f is twice differentiable a Taylor series expansion gives

$$Y_\ell \approx -\frac{1}{4\,N_\ell}\sum_{n=1}^{N_\ell} f''\big(\mathbb{E}\big[g\big(Z^{(n)},W\big)\big]\big)\big(\Delta g_1^{(n)} - \Delta g_2^{(n)}\big)^2.$$

By the Central Limit Theorem, $\Delta g_1^{(n)}, \Delta g_2^{(n)} = O(M_\ell^{-1/2})$ and therefore

$$f''\big(\mathbb{E}\big[g\big(Z^{(n)},W\big)\big]\big)\big(\Delta g_1^{(n)} - \Delta g_2^{(n)}\big)^2 = O(M_\ell^{-1}).$$

It follows that $\mathbb{E}[Y_\ell] = O(M_\ell^{-1})$ and $V_\ell = O(M_\ell^{-2})$. For the MLMC theorem, this corresponds to $\alpha = 1$, $\beta = 2$, $\gamma = 1$, so the complexity is $O(\varepsilon^{-2})$.

This antithetic approach to nested simulation has been developed independently by several authors (Haji-Ali 2012, Chen and Liu 2012, Bujok, Hambly and Reisinger 2013).

Haji-Ali (2012) used it in a mean field model for the motion of crowds, in which each person is modelled as a independent agent subject to random forcing and an additional force due to the collective influence of the crowd. This same approach is also relevant to mean field problems which arise in plasma physics (Rosin *et al.* 2014).

Bujok *et al.* (2013) used multilevel nested simulation for a financial credit derivative application. In their case, the function f was piecewise linear, not twice differentiable, and so the rate of variance convergence was slightly lower, with $\beta = 1.5$. However, this is still sufficiently large to achieve an overall complexity which is $O(\varepsilon^{-2})$.

The pricing of American options is one of the big challenges for Monte Carlo methods in computational finance, and Belomestny, Schoenmakers and Dickmann (2013) have developed a multilevel Monte Carlo for this purpose, based on Anderson and Broadie's dual simulation method (Andersen and Broadie 2004) in which a key component at each time-step in the simulation is to estimate a conditional expectation using a number of sub-paths. In their multilevel treatment, Belomestny *et al.* (2013) use the same uniform time-step on all levels of the simulation, so the quantity which changes between different levels of simulation is the number of sub-samples used to estimate the conditional expectation. Hence, their work is another example of nested simulation.

9.2. MIMC treatment

The analysis in the previous section assumes that we can compute $g(Z,W)$ with $O(1)$ cost, but suppose now that W represents a complete Brownian path, and so $g(Z,W)$ cannot be evaluated exactly; it can only be approximated by using some finite number of time-steps. We could continue to use MLMC, and on level ℓ we could use 2^ℓ time-steps. When using the Milstein discretization (giving first-order weak and strong convergence) this would

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

still give $\alpha = 1$, $\beta = 2$. However, we would now have $\gamma = 2$, because the work on successive levels would go up by factor 4×, because of using twice as many time-steps as well as twice as many inner samples. This then leads to an overall MLMC complexity which is $O(\varepsilon^{-2}(\log \varepsilon)^{-2})$.

Instead we can use MIMC, to get back to an optimal complexity of $O(\varepsilon^{-2})$. We now have a pair of level indices (l_1, l_2), with the number of inner samples equal to 2^{ℓ_1} and the number of time-steps proportional to 2^{ℓ_2}. If we use the natural extension of the MLMC estimator to the corresponding MIMC estimator, which means (for $l_1 > 0, l_2 > 0$) using

$$
\begin{aligned}
Y_\ell = N_\ell^{-1} \sum_{n=1}^{N_\ell} \Bigg\{ & f\Bigg(2^{-\ell_1} \sum_{1}^{2^{\ell_1}} g_{l_2}\big(Z^{(n)}, W^{(m,n)}\big)\Bigg) \\
& - \tfrac{1}{2} f\Bigg(2^{-\ell_1+1} \sum_{1}^{2^{\ell_1-1}} g_{l_2}\big(Z^{(n)}, W^{(m,n)}\big)\Bigg) \\
& - \tfrac{1}{2} f\Bigg(2^{-\ell_1+1} \sum_{2^{\ell_1-1}+1}^{2^{\ell}_1} g_{l_2}\big(Z^{(n)}, W^{(m,n)}\big)\Bigg) \\
& - f\Bigg(2^{-\ell_1} \sum_{1}^{2^{\ell_1}} g_{l_2-1}\big(Z^{(n)}, W^{(m,n)}\big)\Bigg) \\
& + \tfrac{1}{2} f\Bigg(2^{-\ell_1+1} \sum_{1}^{2^{\ell_1-1}} g_{l_2-1}\big(Z^{(n)}, W^{(m,n)}\big)\Bigg) \\
& + \tfrac{1}{2} f\Bigg(2^{-\ell_1+1} \sum_{2^{\ell_1-1}+1}^{2^{\ell}_1} g_{l_2-1}\big(Z^{(n)}, W^{(m,n)}\big)\Bigg) \Bigg\}.
\end{aligned}
$$

The subscript on the g terms denotes the level of time-step approximation.

Carrying out the same analysis as before, performing the Taylor series expansion around $\mathbb{E}[g(Z^{(n)}, W)]$, we obtain

$$
\begin{aligned}
Y_\ell \approx -\frac{1}{4\,N_\ell} \sum_{n=1}^{N_\ell} f''\big(\mathbb{E}\big[g\big(Z^{(n)}, W\big)\big]\big) \big\{ & \big(\Delta g_{1,\ell_2}^{(n)} - \Delta g_{2,\ell_2}^{(n)}\big)^2 \\
& - \big(\Delta g_{1,\ell_2-1}^{(n)} - \Delta g_{2,\ell_2-1}^{(n)}\big)^2 \big\}.
\end{aligned}
$$

The difference of squares can be rearranged as

$$
\begin{aligned}
& \big(\Delta g_{1,\ell_2}^{(n)} - \Delta g_{2,\ell_2}^{(n)}\big)^2 - \big(\Delta g_{1,\ell_2-1}^{(n)} - \Delta g_{2,\ell_2-1}^{(n)}\big)^2 \\
& \quad = \big(\big(\Delta g_{1,\ell_2}^{(n)} + \Delta g_{1,\ell_2-1}^{(n)}\big) - \big(\Delta g_{2,\ell_2}^{(n)} + \Delta g_{2,\ell_2-1}^{(n)}\big)\big) \\
& \qquad \times \big(\big(\Delta g_{1,\ell_2}^{(n)} - \Delta g_{1,\ell_2-1}^{(n)}\big) - \big(\Delta g_{2,\ell_2}^{(n)} - \Delta g_{2,\ell_2-1}^{(n)}\big)\big).
\end{aligned}
$$

Due to the Central Limit Theorem, we have

$$\begin{aligned}\Delta g^{(n)}_{1,\ell_2} + \Delta g^{(n)}_{1,\ell_2-1} &= O(2^{-\ell_1/2}),\\ \Delta g^{(n)}_{2,\ell_2} + \Delta g^{(n)}_{2,\ell_2-1} &= O(2^{-\ell_1/2}),\end{aligned}$$

and assuming first-order strong convergence we also have

$$\begin{aligned}\Delta g^{(n)}_{1,\ell_2} - \Delta g^{(n)}_{1,\ell_2-1} &= O(2^{-\ell_1/2-\ell_2}),\\ \Delta g^{(n)}_{2,\ell_2} - \Delta g^{(n)}_{2,\ell_2-1} &= O(2^{-\ell_1/2-\ell_2}).\end{aligned}$$

Combining these results, we obtain

$$\bigl(\Delta g^{(n)}_{1,\ell_2} - \Delta g^{(n)}_{2,\ell_2}\bigr)^2 - \bigl(\Delta g^{(n)}_{1,\ell_2-1} - \Delta g^{(n)}_{2,\ell_2-1}\bigr)^2 = O(2^{-\ell_1-\ell_2}),$$

and therefore $\mathbb{E}[Y_\ell] = O(2^{-\ell_1-\ell_2})$ and $V_\ell = O(2^{-2\ell_1-2\ell_2})$ with a cost per sample which is $O(2^{\ell_1+\ell_2})$. In Theorem 2.2 this corresponds to $\alpha_1 = \alpha_2 = 1$, $\beta_1 = \beta_2 = 2$, and $\gamma_1 = \gamma_2 = 1$, so the overall complexity is $O(\varepsilon^{-2})$.

Following the analysis in Bujok *et al.* (2013), if the function f is continuous and piecewise differentiable, rather than being twice differentiable, then in the MLMC treatment we would get $\beta = 1.5$, and hence an overall complexity which is $O(\varepsilon^{-2.5})$. On the other hand, with MIMC we would have $\beta_1 = \beta_2 = 1.5$ and so the complexity would remain $O(\varepsilon^{-2})$. This illustrates the benefit of the MIMC approach compared to standard MLMC.

10. Other applications

10.1. Markov chains and limiting distributions

In new research, Glynn and Rhee (2014) consider the application of their randomized version of MLMC to Markov chains, and in particular they have addressed the problem of estimating quantities which are expectations with respect to a limiting distribution. In their application, the Markov chain $\{X_n\}$ in a metric space with metric d is defined by

$$X_0 = x, \quad X_{n+1} = \phi_n(X_n),\ n \geq 0,$$

where $\{\phi_n\}$ is a sequence of i.i.d. random functions. Furthermore, it is assumed that the ϕ are contracting on average in the sense that

$$\sup_{x\neq y} \mathbb{E}\left[\left(\frac{d(\phi_n(x),\phi_n(y))}{d(x,y)}\right)^{2\gamma}\right] < 1$$

for some $\gamma \in (0,1)$. Under these conditions, it is known that the distribution of X_n converges weakly to that of a limit random variable X_∞, and their objective is to estimate expectations of the form $\mathbb{E}[f(X_\infty)]$, where f is Hölder-continuous with exponent γ, so that $|f(y) - f(x)| \leq d(x,y)^\gamma$.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

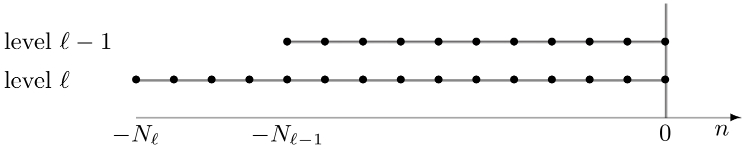

Figure 10.1. Level ℓ and $\ell-1$ paths for contracting Markov chains.

An example they offer of a chain satisfying the required conditions is

$$X_0=0,\quad X_{n+1}=\tfrac{1}{2}X_n+\xi_n,\ n\geq 0,$$

where $\mathbb{P}(\xi_n=0)=\mathbb{P}(\xi_n=1)=1/2$. The invariant distribution in this case is the uniform distribution on $[0,2]$.

In the standard MC approach, one would approximate $\mathbb{E}[f(X_\infty)]$ by estimating $\mathbb{E}[f(X_N)]$ for some large value of N, but this would be a biased estimate. Glynn and Rhee (2014) circumvent this problem by making N increase with level, starting a level ℓ simulation at $n=-N_\ell$ and terminating it at $n=0$, as illustrated in Figure 10.1. The fact that the different levels terminate at the same time is crucial to the multilevel coupling: the level ℓ and $\ell-1$ simulations share the same random ϕ_n for $n\geq -N_{\ell-1}$. Because of the contraction property, the effect of the initial evolution of the level ℓ path for $n<-N_{\ell-1}$ decays exponentially. This gives a coupling with a multilevel correction variance which decays as ℓ increases. Because the decay is exponential in $N_\ell-N_{\ell-1}$, it is appropriate to choose N_ℓ to increase linearly with level.

A very similar approach can also be used for contracting SDEs which converge to a limiting distribution. For these, the level ℓ path will perform a simulation for the time interval $[-T_\ell,0]$, using time-step h_ℓ. The coarse and fine paths will share the same driving Brownian path for the overlapping time interval $[-T_{\ell-1},0]$, and the contraction property will ensure that the multilevel variance decays with level.

10.2. Variable-precision arithmetic

In the latest CPUs from Intel and AMD, each core has a vector unit which can perform eight single-precision or four double-precision operations with one instruction. Hence, single-precision computations can be twice as fast as double-precision on CPUs, and the difference can be even greater on GPUs.

As a very simple example of a two-level MLMC application, Giles (2013) suggested using single-precision arithmetic for samples on level 0 and double-precision arithmetic on level 1. Estimating V_0 and V_1 as defined in Section 1.2 will automatically lead to the optimal allocation of computational effort between the two levels.

This approach has been generalized in new research (Brugger *et al.* 2014). which exploits FPGAs (field-programmable gate arrays) which can perform

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

computations with a user-specified number of bits to represent floating-point or fixed-point numbers. Thus, it is possible to implement a multilevel treatment in which the number of bits used increases with level.

The primary challenge is to ensure that the telescoping sum is correctly respected. In their current work, Brugger *et al.* (2014) use full precision in the generation of the random numbers for the Brownian increments, and then perform the rest of the path calculations with the appropriate number of bits, before doing the final summations in full accuracy. This ensures that the Brownian increments computed for the coarser path $\ell-1$ by summing the increments of the finer path ℓ are consistent with the increments which would be generated on level $\ell-1$ when it is the finer of the two levels.

This would not be the case if the increments on level ℓ were generated with B_ℓ bits of accuracy, then summed to give increments for level $\ell-1$, regardless of whether the truncation to the lower accuracy $B_{\ell-1}$ took place before or after the summation. One possible way of correctly generating and using reduced accuracy Brownian increments would be to use a Brownian bridge construction. Random numbers can be generated in steps by

$$I_n \rightarrow U_n \rightarrow Z_n,$$

where I_n is a random integer on a range $[0, I_{\max}]$, $U_n = (I_n + 1/2)/I_{\max}$ is a random, approximately uniformly distributed variable on the interval $(0,1)$, and $Z_n = \Phi^{-1}(U_n)$ is the corresponding $N(0,1)$ random variable. When using a Brownian bridge construction, as long as the I_n are generated in exactly the same way for each level, the other two steps can be performed with the level of accuracy appropriate to the level of the path, that is, with higher precision on level ℓ than on level $\ell-1$. The Brownian path being generated for level ℓ is then exactly the same, regardless of whether it is the finer or coarser of the two levels being simulated for a particular multilevel correction. Hence, the telescoping sum will be respected.

11. Conclusions

In the last few years there has been considerable progress in the theoretical development, application and analysis of multilevel Monte Carlo methods. On the theoretical side, the key extensions are to unbiased randomized estimators for applications with a rapid rate of variance convergence; Richardson–Romberg extrapolation for improved computational efficiency when the rate of variance convergence is low and the weak error has a regular expansion; and multi-index Monte Carlo (MIMC), generalizing multilevel to multiple ‘directions’ in which approximations can be refined. On the practical side, the range of applications is growing steadily, now including SDEs, Lévy processes, SPDEs, continuous-time Markov processes, nested simulation, and variable-precision arithmetic. Finally, on the numerical analysis

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

side, there has been excellent progress on the variance analysis for the full range of applications, as well as for multilevel QMC algorithms.

This review has attempted to emphasize the conceptual simplicity of the multilevel approach; in essence it is simply a recursive control variate strategy, using cheap inaccurate approximations to some random output quantity as a control variate for more accurate but more costly approximations.

In practice, the challenge is to develop a tight coupling between successive approximation levels, to minimize the variance of the difference in the output obtained from each level. One particular difficulty which can arise in some applications is when the output quantity is a discontinuous function of the intermediate solution, as in the case of a digital option associated with a Brownian diffusion SDE. Three treatments were described to effectively smooth the output to improve the variance in such cases: a analytic conditional expectation, a 'splitting' approximation, and a change of measure. Similar treatments have been found to be helpful in other contexts.

It was also shown that techniques such as antithetic variates can be used to improve the variance of the multilevel estimator without improving the strong convergence properties of the numerical discretization.

Looking to the future, there are many directions for further research. Some areas to perhaps highlight are as follows.

- Development and application of algorithms for MLQMC.

There has been excellent recent work on the theoretical analysis of MLQMC, but this has not yet been matched by the development and application of improved algorithms.

- New applications using MIMC.

MIMC, the multi-index Monte Carlo method, is probably the most exciting new theoretical development in this area, and it offers considerable scope for improved computational efficiency in a range of applications.

- More work on sensitivity analysis.

There has been very little research so far on MLMC for sensitivity analysis, but in some application areas such as mathematical finance this is very important.

- Work on combinations with other variance reduction techniques, such as Latin hypercube or importance sampling.

It is often helpful to combine different variance reduction approaches to obtain greater overall savings, and it seems likely that MLMC can be helpfully combined with other techniques.

- Nested simulation / mean field games.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

This looks like a very promising new application area for both MLMC and MIMC. Such simulations can be very expensive using traditional Monte Carlo methods, so MLMC/MIMC may give very substantial computational savings.

The MATLAB codes which produced all of the results in this article are available from Giles (2014), along with a number of additional MLMC application codes demonstrating adaptive time-stepping for SDEs and continuous-time Markov processes, the use of antithetics for multi-dimensional SDEs, and additional SPDE examples.

For further information on multilevel Monte Carlo methods, readers can refer to http://people.maths.ox.ac.uk/gilesm/mlmc_community.html, which lists the research groups working in the area, and their main publications.

Acknowledgements

The author is very grateful to Klaus Ritter and Raúl Tempone for their feedback on a draft of this article. Much of the content is the outcome of collaborations and interactions with many researchers, too many to list here but all much appreciated.

REFERENCES[3]

L. Andersen and M. Broadie (2004), 'A primal–dual simulation algorithm for pricing multi-dimensional American options', *Management Sci.* **50**, 1222–1234.

D. Anderson and D. Higham (2012), 'Multi-level Monte Carlo for continuous time Markov chains with applications in biochemical kinetics', *SIAM Multiscale Model. Simul.* **10**, 146–179.

D. Anderson, D. Higham and Y. Sun (2014), 'Complexity of multilevel Monte Carlo tau-leaping', *SIAM J. Numer. Anal.* **52**, 3106–3127.

A. Asmussen and P. Glynn (2007), *Stochastic Simulation*, Springer.

R. Avikainen (2009), 'On irregular functionals of SDEs and the Euler scheme', *Finance Stoch.* **13**, 381–401.

I. Babuška, F. Nobile and R. Tempone (2010), 'A stochastic collocation method for elliptic partial differential equations with random input data', *SIAM Rev.* **52**, 317–355.

I. Babuška, R. Tempone and G. Zouraris (2004), 'Galerkin finite element approximations of stochastic elliptic partial differential equations', *SIAM J. Numer. Anal.* **42**, 800–825.

J. Baldeaux and M. Gnewuch (2014), 'Optimal randomized multilevel algorithms for infinite-dimensional integration on function spaces with ANOVA-type decomposition', *SIAM J. Numer. Anal.* **52**, 1128–1155.

[3] The URLs cited in this work were correct at the time of going to press, but the publisher and the authors make no undertaking that the citations remain live or are accurate or appropriate.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

A. Barth, A. Lang and C. Schwab (2013), 'Multilevel Monte Carlo method for parabolic stochastic partial differential equations', *BIT Numer. Math.* **53**, 3–27.

A. Barth, C. Schwab and N. Zollinger (2011), 'Multi-level Monte Carlo finite element method for elliptic PDEs with stochastic coefficients', *Numer. Math.* **119**, 123–161.

D. Belomestny, J. Schoenmakers and F. Dickmann (2013), 'Multilevel dual approach for pricing American style derivatives', *Finance Stoch.* **17**, 717–742.

A. Brandt and V. Ilyin (2003), 'Multilevel Monte Carlo methods for studying large scale phenomena in fluids', *J. Molecular Liquids* **105**, 245–248.

A. Brandt, M. Galun and D. Ron (1994), 'Optimal multigrid algorithms for calculating thermodynamic limits', *J. Statist. Phys.* **74**, 313–348.

M. Broadie and P. Glasserman (1996), 'Estimating security price derivatives using simulation', *Management Sci.* **42**, 269–285.

C. Brugger, C. de Schryver, N. Wehn, S. Omland, M. Hefter, K. Ritter, A. Kostiuk and R. Korn (2014), Mixed precision multilevel Monte Carlo on hybrid computing systems. In *Proc. Conference on Computational Intelligence for Financial Engineering and Economics*, IEEE.

K. Bujok and C. Reisinger (2012), 'Numerical valuation of basket credit derivatives in structural jump-diffusion models', *J. Comput. Finance* **15**, 115–158.

K. Bujok, B. Hambly and C. Reisinger (2013), 'Multilevel simulation of functionals of Bernoulli random variables with application to basket credit derivatives', *Methodol. Comput. Appl. Probab.* doi:10.1007/s11009-013-9380-5

H.-J. Bungartz and M. Griebel (2004), Sparse grids. In *Acta Numerica*, Vol. 13, Cambridge University Press, pp. 147–269.

S. Burgos (2014), The computation of Greeks with multilevel Monte Carlo. DPhil thesis, University of Oxford.

S. Burgos and M. Giles (2012), Computing Greeks using multilevel path simulation. In *Monte Carlo and Quasi-Monte Carlo Methods 2010* (L. Plaskota and H. Woźniakowski, eds), Springer, pp. 281–296.

J. Charrier, R. Scheichl and A. Teckentrup (2013), 'Finite element error analysis of elliptic PDEs with random coefficients and its application to multilevel Monte Carlo methods', *SIAM J. Numer. Anal.* **51**, 322–352.

N. Chen and Y. Liu (2012), Estimating expectations of functionals of conditional expected via multilevel nested simulation. Presentation at conference on Monte Carlo and Quasi-Monte Carlo Methods, Sydney.

J. Clark and R. Cameron (1980), The maximum rate of convergence of discrete approximations for stochastic differential equations. In *Stochastic Differential Systems Filtering and Control* (B. Grigelionis, ed.), Vol. 25 of *Lecture Notes in Control and Information Sciences*, Springer, pp. 162–171.

K. Cliffe, M. Giles, R. Scheichl and A. Teckentrup (2011), 'Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients', *Comput. Visualization Sci.* **14**, 3–15.

N. Collier, A.-L. Haji-Ali, F. Nobile, E. von Schwerin and R. Tempone (2014), 'A continuation multilevel Monte Carlo algorithm', *BIT Numer. Math.* doi:10.1007/s10543-014-0511-3

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

J. Creutzig, S. Dereich, T. Müller-Gronbach and K. Ritter (2009), 'Infinite-dimensional quadrature and approximation of distributions', *Found. Comput. Math.* **9**, 391–429.

T. Daun and S. Heinrich (2013), Complexity of Banach space valued and parametric integration. In *Monte Carlo and Quasi-Monte Carlo Methods 2012* (J. Dick, F. Kuo, G. Peters and I. Sloan, eds), Springer, pp. 297–316.

T. Daun and S. Heinrich (2014*a*), 'Complexity of parametric initial value problems in Banach spaces', *J. Complexity* **30**, 392–429.

T. Daun and S. Heinrich (2014*b*), 'Complexity of parametric integration in various smoothness classes', *J. Complexity* **30**, 750–766.

S. Dereich (2011), 'Multilevel Monte Carlo algorithms for Lévy-driven SDEs with Gaussian correction', *Ann. Appl. Probab.* **21**, 283–311.

S. Dereich and F. Heidenreich (2011), 'A multilevel Monte Carlo algorithm for Lévy-driven stochastic differential equations', *Stochastic Process. Appl.* **121**, 1565–1587.

S. Dereich, A. Neuenkirch and L. Szpruch (2012), 'An Euler-type method for the strong approximation of the Cox–Ingersoll–Ross process', *Proc. Royal Soc. A* **468**(2140), 1105–1115.

J. Dick and M. Gnewuch (2014*a*), 'Infinite-dimensional integration in weighted Hilbert spaces: Anchored decompositions, optimal deterministic algorithms, and higher-order convergence', *Found. Comput. Math.* **14**, 1027–1077.

J. Dick and M. Gnewuch (2014*b*), 'Optimal randomized changing dimension algorithms for infinite-dimensional integration on function spaces with ANOVA-type decomposition', *J. Approx. Theory* **184**, 111–145.

J. Dick, F. Kuo and I. Sloan (2013), High-dimensional integration: The quasi-Monte Carlo way. In *Acta Numerica*, Vol. 22, Cambridge University Press, pp. 133–288.

J. Dick, F. Pillichshammer and B. Waterhouse (2007), 'The construction of good extensible rank-1 lattices', *Math. Comp.* **77**, 2345–2373.

C. Dietrich and G. Newsam (1997), 'Fast and exact simulation of stationary Gaussian processes through circulant embedding of the covariance matrix', *SIAM J. Sci. Comput.* **18**, 1088–1107.

Y. Efendiev, O. Iliev and C. Kronsbein (2013), 'Multilevel Monte Carlo methods using ensemble level mixed MsFEM for two-phase flow and transport simulations', *Comput. Geosci.* **17**, 833–850.

A. Ferreiro-Castilla, A. Kyprianou, R. Scheichl and G. Suryanarayana (2014), 'Multi-level Monte-Carlo simulation for Lévy processes based on the Wiener–Hopf factorisation', *Stochastic Process. Appl.* **124**, 985–1010.

M. Giles (2008*a*), Improved multilevel Monte Carlo convergence using the Milstein scheme. In *Monte Carlo and Quasi-Monte Carlo Methods 2006* (A. Keller, S. Heinrich and H. Niederreiter, eds), Springer, pp. 343–358.

M. Giles (2008*b*), 'Multilevel Monte Carlo path simulation', *Operations Research* **56**, 607–617.

M. Giles (2009), Multilevel Monte Carlo for basket options. In *Proc. 2009 Winter Simulation Conference* (M. Rossetti, R. Hill, B. Johansson, A. Dunkin and R. Ingalls, eds), IEEE, pp. 1283–1290.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

M. Giles (2013), Multilevel Monte Carlo methods. In *Monte Carlo and Quasi-Monte Carlo Methods 2012* (J. Dick, F. Kuo, G. Peters and I. Sloan, eds), Springer, pp. 79–98.

M. Giles (2014), MATLAB code for multilevel Monte Carlo computations. http://people.maths.ox.ac.uk/gilesm/acta/

M. Giles and C. Reisinger (2012), 'Stochastic finite differences and multilevel Monte Carlo for a class of SPDEs in finance', *SIAM J. Financial Math.* **3**, 572–592.

M. Giles and L. Szpruch (2013), Antithetic multilevel Monte Carlo estimation for multidimensional SDEs. In *Monte Carlo and Quasi-Monte Carlo Methods 2012* (J. Dick, F. Kuo, G. Peters and I. Sloan, eds), Springer, pp. 367–384.

M. Giles and L. Szpruch (2014), 'Antithetic multilevel Monte Carlo estimation for multi-dimensional SDEs without Lévy area simulation', *Ann. Appl. Probab.* **24**, 1585–1620.

M. Giles and B. Waterhouse (2009), Multilevel quasi-Monte Carlo path simulation. In *Advanced Financial Modelling*, Radon Series on Computational and Applied Mathematics, De Gruyter, pp. 165–181.

M. Giles, K. Debrabant and A. Rößler (2013), Numerical analysis of multilevel Monte Carlo path simulation using the Milstein discretisation. arXiv:1302.4676

M. Giles, D. Higham and X. Mao (2009), 'Analysing multilevel Monte Carlo for options with non-globally Lipschitz payoff', *Finance Stoch.* **13**, 403–413.

M. Giles, C. Lester and J. Whittle (2015*a*), Simple adaptive timestepping for multilevel Monte Carlo. In *Monte Carlo and Quasi-Monte Carlo Methods 2014*, Springer, to appear.

M. Giles, T. Nagapetyan and K. Ritter (2015*b*), 'Multilevel Monte Carlo approximation of distribution functions and densities', *SIAM/ASA J. Uncertainty Quantification*, to appear.

D. Gillespie (1976), 'A general method for numerically simulating the stochastic time evolution of coupled chemical reactions', *J. Comput. Phys.* **22**, 403–434.

P. Glasserman (2004), *Monte Carlo Methods in Financial Engineering*, Springer.

P. Glasserman and N. Merener (2004), 'Convergence of a discretization scheme for jump-diffusion processes with state-dependent intensities', *Proc. Royal Soc. London A* **460**, 111–127.

P. Glynn and C.-H. Rhee (2014), 'Exact estimation for Markov chain equilibrium expectations', *J. Appl. Probab.* **51A**, 377–389.

E. Gobet and S. Menozzi (2010), 'Stopped diffusion processes: overshoots and boundary correction', *Stochastic Process. Appl.* **120**, 130–162.

S. Graubner (2008), Multi-level Monte Carlo Methoden für stochastische partielle Differentialgleichungen. Diplomarbeit, TU Darmstadt.

M. Gunzburger, C. Webster and G. Zhang (2014), Stochastic finite element methods for partial differential equations with random input data. In *Acta Numerica*, Vol. 23, Cambridge University Press, pp. 521–650.

A.-L. Haji-Ali (2012), Pedestrian flow in the mean-field limit. MSc thesis, KAUST.

A.-L. Haji-Ali, F. Nobile and R. Tempone (2014*a*), Multi Index Monte Carlo: When sparsity meets sampling. arXiv:1405.3757

A.-L. Haji-Ali, F. Nobile, E. von Schwerin and R. Tempone (2014*b*), Optimization of mesh hierarchies in multilevel Monte Carlo samplers. arXiv:1403.2480

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

S. Heinrich (1998), 'Monte Carlo complexity of global solution of integral equations', *J. Complexity* **14**, 151–175.

S. Heinrich (2000), The multilevel method of dependent tests. In *Advances in Stochastic Simulation Methods* (N. Balakrishnan, V. Melas and S. Ermakov, eds), Springer, pp. 47–61.

S. Heinrich (2001), Multilevel Monte Carlo methods. In *Multigrid Methods*, Vol. 2179 of *Lecture Notes in Computer Science*, Springer, pp. 58–67.

S. Heinrich (2006), 'Monte Carlo approximation of weakly singular integral operators', *J. Complexity* **22**, 192–219.

S. Heinrich and E. Sindambiwe (1999), 'Monte Carlo complexity of parametric integration', *J. Complexity* **15**, 317–341.

D. Higham, X. Mao and A. Stuart (2002), 'Strong convergence of Euler-type methods for nonlinear stochastic differential equations', *SIAM J. Numer. Anal.* **40**, 1041–1063.

D. Higham, X. Mao, M. Roj, Q. Song and G. Yin (2013), 'Mean exit times and the multi-level Monte Carlo method', *SIAM J. Uncertainty Quantification* **1**, 2–18.

H. Hoel, E. von Schwerin, A. Szepessy and R. Tempone (2012), Adaptive multilevel Monte Carlo simulation. In *Numerical Analysis of Multiscale Computations*, (B. Engquist, O. Runborg and Y.-H. Tsai, eds), Vol. 82 of *Lecture Notes in Computational Science and Engineering*, Springer, pp. 217–234.

H. Hoel, E. von Schwerin, A. Szepessy and R. Tempone (2014), 'Implementation and analysis of an adaptive multilevel Monte Carlo algorithm', *Monte Carlo Methods Appl.* **20**, 1–41.

M. Hutzenthaler, A. Jentzen and P. Kloeden (2013), 'Divergence of the multilevel Monte Carlo method', *Ann. Appl. Probab.* **23**, 1913–1966.

O. Iliev, T. Nagapetyan and K. Ritter (2013), Monte Carlo simulation of asymmetric flow field flow fractionation. In *Monte Carlo Methods and Applications: Proc. 8th IMACS Seminar on Monte Carlo Methods*, de Gruyter, pp. 115–123.

S. Joe and F. Kuo (2008), 'Constructing Sobol sequences with better two-dimensional projections', *SIAM J. Sci. Comput.* **30**, 2635–2654.

A. Kebaier (2005), 'Statistical Romberg extrapolation: A new variance reduction method and applications to options pricing', *Ann. Appl. Probab.* **14**, 2681–2705.

A. Kebaier and A. Kohatsu-Higa (2008), 'An optimal control variance reduction method for density estimation', *Stochastic Process. Appl.* **118**, 2143–2180.

P. Kloeden and E. Platen (1992), *Numerical Solution of Stochastic Differential Equations*, Springer.

F. Kuo, C. Schwab and I. Sloan (2015), 'Multi-level quasi-Monte Carlo finite element methods for a class of elliptic partial differential equations with random coefficients'. *Found. Comput. Math.* **15**, 411–449.

T. Kurtz (1982), Representation and approximation of counting processes. In *Advances in Filtering and Optimal Stochastic Control*, Vol. 42, Springer, pp. 177–191.

P. L'Ecuyer (1990), 'A unified view of the IPA SF and LR gradient estimation techniques', *Management Sci.* **36**, 1364–1383.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

M. Ledoux and M. Talagrand (1991), *Probability in Banach Spaces: Isoperimetry and Processes*, Springer.

V. Lemaire and G. Pagès (2013), Multilevel Richardson–Romberg extrapolation. arXiv:1401.1177

C. Lester, C. Yates, M. Giles and R. Baker (2015), 'An adapted multi-level simulation algorithm for stochastic biological systems', *J. Chem. Phys.*, to appear.

Q. Li (2007), Numerical approximation for SDE. PhD thesis, University of Edinburgh.

H. Marxen (2010), 'The multilevel Monte Carlo method used on a Lévy driven SDE', *Monte Carlo Methods Appl.* **16**, 167–190.

R. Merton (1976), 'Option pricing when underlying stock returns are discontinuous', *J. Finance* **3**, 125–144.

R. Mikulevicius and E. Platen (1988), 'Time discrete Taylor approximations for Itô processes with jump component', *Mathematische Nachrichten* **138**, 93–104.

S. Mishra, C. Schwab and J. Sukys (2012), 'Multi-level Monte Carlo finite volume methods for nonlinear systems of conservation laws in multi-dimensions', *J. Comput. Phys.* **231**, 3365–3388.

A. Moraes, R. Tempone and P. Vilanova (2014), A multilevel adaptive reaction-splitting simulation method for stochastic reaction networks. arXiv:1406.1989

A. Moraes, R. Tempone and P. Vilanova (2015), 'Multilevel hybrid Chernoff tau-leap', *BIT Numer. Math.*, to appear.

F. Müller, P. Jenny and D. Meyer (2013), 'Multilevel Monte Carlo for two phase flow and Buckley–Leverett transport in random heterogeneous porous media', *J. Comput. Phys.* **250**, 685–702.

B. Niu, F. Hickernell, T. Müller-Gronbach and K. Ritter (2010), 'Deterministic multi-level algorithms for infinite-dimensional integration on $\mathbb{R}^N$', *J. Complexity* **27**, 331–351.

E. Platen and N. Bruti-Liberati (2010), *Numerical Solution of Stochastic Differential Equations With Jumps in Finance*, Springer.

T. Primozic (2011), Estimating expected first passage times using multilevel Monte Carlo algorithm. MSc thesis, University of Oxford.

M. Putko, A. Taylor, P. Newman and L. Green (2002), 'Approach for input uncertainty propagation and robust design in CFD using sensitivity derivatives', *J. Fluids Engng* **124**, 60–69.

C.-H. Rhee and P. Glynn (2012), A new approach to unbiased estimation for SDEs. In *Proc. 2012 Winter Simulation Conference*, IEEE.

C.-H. Rhee and P. Glynn (2015), 'Unbiased estimation with square root convergence for SDE models', *Operations Research*, to appear.

M. Rosin, L. Ricketson, A. Dimits, R. Caflisch and B. Cohen (2014), 'Multilevel Monte Carlo simulation of Coulomb collisions', *J. Comput. Phys.* **247**, 140–157.

W. Schoutens (2003), *Lévy Processes in Finance: Pricing Financial Derivatives*, Wiley.

B. Silverman (1986), *Density Estimation for Statistics and Data Analysis*, Chapman & Hall.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press

A. Speight (2009), ‘A multilevel approach to control variates’, *J. Comput. Finance* **12**, 1–25.

A. Teckentrup (2013), Multilevel Monte Carlo methods and uncertainty quantification. PhD thesis, University of Bath.

A. Teckentrup, R. Scheichl, M. Giles and E. Ullmann (2013), ‘Further analysis of multilevel Monte Carlo methods for elliptic PDEs with random coefficients’, *Numer. Math.* **125**, 569–600.

F. Vidal-Codina, N. Nguyen, M. Giles and J. Peraire (2014), A model and variance reduction method for computing statistical outputs of stochastic elliptic partial differential equations. arXiv:1409.1526

Y. Xia (2014), Multilevel Monte Carlo for jump processes. DPhil thesis, University of Oxford.

Y. Xia and M. Giles (2012), Multilevel path simulation for jump-diffusion SDEs. In *Monte Carlo and Quasi-Monte Carlo Methods 2010* (L. Plaskota and H. Woźniakowski, eds), Springer, pp. 695–708.

Y. Xia and M. Giles (2014), Multilevel Monte Carlo for exponential Lévy models. arXiv:1403.5309

D. Xiu and G. Karniadakis (2002), ‘The Wiener–Askey polynomial chaos for stochastic differential equations’, *SIAM J. Sci. Comput.* **24**, 619–644.

https://doi.org/10.1017/S096249291500001X Published online by Cambridge University Press