An Introduction to Multilevel Monte Carlo for Option Valuation

Desmond J. Higham*

June 26, 2015

Abstract

Monte Carlo is a simple and flexible tool that is widely used in computational finance. In this context, it is common for the quantity of interest to be the expected value of a random variable defined via a stochastic differential equation. In 2008, Giles proposed a remarkable improvement to the approach of discretizing with a numerical method and applying standard Monte Carlo. His *multilevel Monte Carlo* method offers a speed up of $O(\epsilon^{-1})$, where ϵ is the required accuracy. So computations can run 100 times more quickly when two digits of accuracy are required. The "multilevel philosophy" has since been adopted by a range of researchers and a wealth of practically significant results has arisen, most of which have yet to make their way into the expository literature. In this work, we give a brief, accessible, introduction to multilevel Monte Carlo and summarize recent results applicable to the task of option evaluation.

Keywords computational complexity, control variate, Euler–Maruyama, Monte Carlo, option value, stochastic differential equation, variance reduction.

1   Aims

Finding the appropriate market value of a financial option can usually be formulated as an expected value computation [23, 26, 41]. In the case where the product underlying the option is modelled as a stochastic differential equation (SDE), we may

---

*Department of Mathematics and Statistics, University of Strathclyde, UK

- simulate the SDE numerically to compute many independent sample paths, and then

- combine the option payoff from each path in order to obtain a Monte Carlo estimate, and an accompanying confidence interval.

Compared with other approaches, notably the direct discretization of a partial differential equation based formulation of the problem, a Monte Carlo computation has the advantages of (a) being simple to implement and (b) being flexible enough to cope with a wide range of underlying SDE models and option payoffs. On the downside, Monte Carlo is typically expensive in terms of computation time [23, 26].

In the seminal 2008 paper [17], Giles pulled together ideas from numerical analysis, stochastic analysis and applied statistics in order to deliver a dramatic improvement on the efficiency of the "SDE simulation plus Monte Carlo" approach. If the required level of accuracy, in terms of confidence interval, is ϵ, the multilevel approach essentially improves the computational complexity by a factor of ϵ. So for a calculation requiring two digits of accuracy, we obtain a hundredfold improvement in computation time. Multilevel Monte Carlo has rapidly become an extremely hot topic in the field of stochastic computation, impacting on a wide range of application areas. In particular, technical reviews of research progress in the field have begun to appear [19, 21] and a comprehensive survey is currently in progress by Giles for the journal Acta Numerica. However, the area is still sufficiently new that most textbooks in computational finance do not introduce the topic, and hence it has not been fully integrated into typical graduate-level classes and development courses for practitioners. For this reason, we present here an accessible introduction to the multilevel Monte Carlo approach, and give a brief overview of the current state of the art with respect to financial option valuation.

In section 2 we discuss the underlying SDE simulation. Section 3 then considers the complexity of standard Monte Carlo in this setting. In section 4 we give some motivation for the multilevel approach, which is introduced and analysed in section 5. Section 6 illustrates the performance of the algorithm in practice, using code that has been made available by Giles. In section 7 we give pointers to multilevel research in option valuation that has built on [17]. Section 8 concludes with a brief discussion.

2 Convergence in SDE Simulation

We consider an Ito SDE of the form

$$dX(t) = f(X(t))dt + g(X(t))dW(t), \quad X(0) = X_0. \qquad (1)$$

Here, $f : \mathbb{R}^m \to \mathbb{R}^m$ and $g : \mathbb{R}^m \to \mathbb{R}^{m \times d}$ are given functions, known as the drift and diffusion coefficients, respectively, and $W(t) \in \mathbb{R}^d$ is standard Brownian motion. The initial condition X_0 is supplied and we wish to simulate the SDE over the fixed time interval $[0, T]$. The Euler–Maruyama method [34, 36] computes approximations $X_n \approx X(t_n)$, where $t_n = n\Delta t$, according to $X_0 = X(0)$ and, for $n = 1, 2, \ldots N - 1$,

$$X_{n+1} = X_n + f(X_n)\Delta t + g(X_n)\Delta W_n, \qquad (2)$$

where $\Delta t = T/N$ is the stepsize and $\Delta W_n = W(t_{n+1}) - W(t_n)$ is the relevant Brownian motion increment.

In the study of the accuracy of SDE simulation methods, the two most widely used convergence concepts are referred to as *weak* and *strong* [34, 36]. Roughly,

- weak convergence controls the error of the means, whereas,

- strong convergence controls the mean of the error.

To prove weak and strong convergence results, we must impose conditions on the SDE. For example it is standard to assume that f and g in (1) satisfy global Lipschitz conditions; that is, there exists a constant $\widehat{L}$ such that

$$|f(x) - f(y)| \le \widehat{L}|x - y| \quad \text{and} \quad |g(x) - g(y)| \le \widehat{L}|x - y|, \quad \text{for all } x, y \in \mathbb{R}^m. \qquad (3)$$

Here and throughout we take $\|\cdot\|$ to be the Euclidean norm. Under such conditions, and for appropriate initial data, it follows that the Euler–Maruyama method has weak order one, so that

$$\sup_{0 \le t_n \le T} (\mathbb{E}[X(t_n)] - \mathbb{E}[X_n]) = O(\Delta t). \qquad (4)$$

In the sense of strong error, which involves the mean of the absolute difference between the two random variables at each grid point, Euler–Maruyama achieves only an order of one half in general:

$$\mathbb{E}\left[\sup_{0 \le t_n \le T} |X(t_n) - X_n| \right] = O(\Delta t^{\frac{1}{2}}). \qquad (5)$$

3

More generally, for any $m > 1$ and sufficiently small Δt there is a constant $C = C(m)$ such that

$$\mathbb{E}\left[\sup_{0 \le t_n \le T} |X(t_n) - X_n|^m\right] \le C\Delta t^{m/2}. \tag{6}$$

Strong convergence is sometimes described as a pathwise property. This can be understood via the Borel-Cantelli Lemma. For example, in [33] it is shown that given any $\epsilon > 0$ there exists a path-dependent constant $K = K(\epsilon)$ such that, for all sufficiently small Δt,

$$\sup_{0 \le t_n \le T} |X(t_n) - X_n| \le K(\epsilon)\Delta t^{\frac{1}{2}-\epsilon}.$$

In the setting of this work, it is tempting to argue that strong convergence is not relevant; if we wish to compute an expected value based on the SDE solution then following individual paths accurately is not important. However, we will see in section 5 that the analysis in [17] justifying multilevel Monte Carlo makes use of both weak and strong convergence properties.

To conclude this section, we remark that the analysis of SDE simulation on problems that violate the global Lipschitz conditions (3) is far from complete. In the case of SDE models for financial assets and interest rates, issues may arise through faster than linear growth at infinity and also through unbounded derivatives at the origin. For example, both complications occur in the class of scalar interest rate models from [1],

$$dX(t) = \left(\alpha_{-1}X(t)^{-1} - \alpha_0 + \alpha_1 X(t) - \alpha_2 X(t)^r\right) dt + \alpha_3 X(t)^\rho dW(t),$$

where the α_i are positive constants and $r, \rho > 1$. Although some positive results are available for specific nonlinear structures [27, 28, 29, 42], there has also been a sequence of negative results showing how Euler–Maruyama can break down on nonlinear SDEs [27, 30, 37].

3 SDE Simulation and Standard Monte Carlo

Given the SDE (1), suppose we wish to approximate the final time expected value of the solution, $\mathbb{E}[X(T)]$, using Monte Carlo with Euler–Maruyama. We will let ϵ denote the accuracy requirement in terms of confidence interval width; fixing on a 95% confidence level to be concrete, we therefore wish to be in a position where applying the algorithm independently a large number of times, the exact answer would be within $\pm\epsilon$ of our computed answer with frequency at least 0.95.

Let $X_N^{[s]}$ denote the Euler–Maruyama final time approximation along the sth path. Using M Monte Carlo samples we may form the sample average

$$a_M = \frac{1}{M} \sum_{s=1}^{M} X_N^{[s]}.$$

The overall error in our approximation has the form

$$
\begin{aligned}
a_M - \mathbb{E}[X(T)] &= a_M - \mathbb{E}[X(T) - X_N + X_N] \\
&= a_M - \mathbb{E}[X_N] + \mathbb{E}[X_N - X(T)]. \quad (7)
\end{aligned}
$$

Note that X_N denotes a random variable describing the result of applying Euler–Maruyama (2), whereas each $X_N^{[s]}$ is an independent sample from the distribution given by X_N. The expression (7) breaks down the error into two terms. The statistical error, $a_M - \mathbb{E}[X_n]$, is concerned with how well we can approximate an expected value from a finite number of samples; it does not depend on how accurately the numerical method approximates the SDE (in particular it does not depend significantly on Δt) and it will generally decrease if we take more sample paths. The discretization error, or bias, $\mathbb{E}[X_N - X(T)]$, arises because we have approximated the SDE with a difference equation; this is the discrepancy that would remain if we had access to the exact expected value of the numerical solution and it will generally decrease if we reduce the stepsize.

Standard results [23, 40] tell us that the statistical error $a_M - \mathbb{E}[X_n]$ can be described via a confidence interval of width $O(1/\sqrt{M})$. The weak convergence property (4) shows that the bias $\mathbb{E}[X_N - X(T)]$ behaves like $O(\Delta t)$; so we must add this amount to the confidence interval width. We arrive at an overall confidence interval of width $O(1/\sqrt{M}) + O(\Delta t)$. To achieve our required target accuracy of ϵ, we see that $1/\sqrt{M}$ and Δt should scale like ϵ. In other words, M should scale like ϵ^{-2} and Δt should scale like ϵ.

It is reasonable to measure computational cost by counting either the number of times that the drift and diffusion coefficients, f and g, are evaluated, or the number of times that a random number generator is called. In either case, the cost per path is proportional to $1/\Delta t$, and hence the total cost of the computation scales like $M/\Delta t$. We argued above that M should scale like ϵ^{-2} and Δt should scale like ϵ. Here is the conclusion:

> we may achieve accuracy ϵ by combining Euler–Maruyama and standard Monte Carlo at an overall cost that scales like ϵ^{-3}.

One approach to improving the computational complexity is to replace Euler–Maruyama with a simulation method of higher weak order [4, 34, 36].

If we use a second order method, so that (4) is replaced by

$$\sup_{0 \le t_n \le T} (\mathbb{E}[X(t_n)] - \mathbb{E}[X_n]) = O(\Delta t^2),$$

then a straightforward adaption of the arguments above lead to the following conclusion:

> we may achieve accuracy ϵ by combining a second order weak method and standard Monte Carlo at an overall cost that scales like $\epsilon^{-2.5}$.

We note, however, that establishing second order weak convergence requires extra smoothness assumptions to be placed on the SDE coefficients.

As we show in section 5, the method of Giles [17] has the following feature:

> we may achieve accuracy ϵ by using Euler–Maruyama in a multilevel Monte Carlo setting at an overall cost that scales like $\epsilon^{-2}(\log \epsilon)^2$.

Moreover, by using a higher strong order method, such as Milstein [34, 36], it is possible to reduce the multilevel Monte Carlo cost to the order of ϵ^{-2} [16].

It is worth pausing to admire an $O(\epsilon^{-2})$ computational complexity count. Suppose we are given an exact expression for the SDE solution, as a function of $W(t)$. Hence, we are able to compute exact samples, without the need to apply a numerical method. A standard Monte Carlo approach requires $1/\sqrt{M}$ to scale like ϵ in order to achieve the required confidence interval width. If we regard the evaluation of each exact $X(T)$ sample as having $O(1)$ cost, then the cost overall will be proportional to M; that is, ϵ^{-2}. In this sense, with a multilevel approach *the numerical analysis comes for free*; we can solve the problem as quickly as one for which we have an exact pathwise expression for the SDE solution.

4 Motivation for the Multilevel Approach

We can motivate the multilevel approach by considering a series expansion of Brownian motion, where the coefficients are random variables. The *Paley-Wiener representation* over $[0, 2\pi]$ has the form

$$W(t) = Z_0 \frac{t}{\sqrt{2\pi}} + \frac{2}{\sqrt{\pi}} \sum_{n=1}^{\infty} Z_n \frac{\sin(\frac{1}{2}nt)}{n}, \tag{8}$$

6

where the $\{Z_i\}_{i \geq 0}$ are i.i.d. and $N(0, 1)$; see, for example, [35]. In Figure 1 we draw samples for the Z_i and plot the curves arising when the infinite series in (8) is truncated to $\sum_{n=1}^{M}$, for $M = 1, 2, 5, 10, 50$ and 200. It is clear that the early terms in the series affect the overall shape, while the later terms add fine detail. From this perspective, it is intuitively reasonable that we can build up information at different resolution scales, with the finer scales having less impact on the overall picture.

Now, we may view Monte Carlo as requiring a "black box" that returns independent samples. In our numerical SDE context, the samples come from a distribution that is only approximately correct, and the black box (the Euler–Maruyama method) comes with a dial. Turning the dial corresponds to changing Δt. Samples with a smaller Δt are more expensive—we have to wait longer for them because the paths contain more steps. The multilevel Monte Carlo algorithm cleverly exploits this dial. The black box is used to produce samples across a range of stepsizes. Most of the samples that we ask for will be obtained quickly with relatively large Δt values. Correspondingly few samples will be generated at the expensive small Δt levels. In a sense, the large Δt paths cover the low-frequency information so that expensive, high-frequency paths are used sparingly. Figure 1 might convince you that this idea has some merit. The next section works through the details.

5   Multilevel Monte Carlo with Euler–Maruyama

We focus now on the more general case where we wish to approximate the expected value of some function of the final time solution, $\mathbb{E}[h(X(T))]$. We have in mind the case where $X(t)$ represents an underlying asset, in risk-neutral form, and $h(\cdot)$ is the payoff of a corresponding European-style option [23, 26]. For example, $h(x) = \max(x - E, 0)$ for a European call option with exercise price E and expiry time T. For simplicity, we will consider the scalar case, so that $m = d = 1$ in (1), but we note that all arguments generalise to the case of systems, with the same conclusions. We assume that the payoff function h satisfies a global Lipschitz condition; this covers the call and put option cases.

Multilevel Monte Carlo uses a range of different discretization levels. At level l we have a stepsize of the form

$$\Delta t_l = K^{-l}T, \quad \text{where } l = 0, 1, 2, \ldots, L. \tag{9}$$

Here $K > 1$ is a fixed quantity whose precise value does not affect the overall complexity of the method, in terms of the asymptotic rate as $\epsilon \to 0$. For

simplicity we may think of $K = 2$. As the upper limit on the level index we choose

$$L = \frac{\log \epsilon^{-1}}{\log K}. \tag{10}$$

In this way, at the coarsest level, $l = 0$, we have the largest stepsize, $\Delta t_0 = T$, covering the whole interval in one step. At the most refined level, $l = L$, we have $\Delta t_L = O(\epsilon)$—from (4), this the stepsize needed by Euler–Maruyama to achieve weak error of $O(\epsilon)$.

With each choice of stepsize, Δt_l, we may apply Euler–Maruyama to the SDE (1) and evaluate the payoff function h at the final time. We will let the random variable $\widehat{P}_l$ denote this approximation to $h(X(T))$. Now, from the linearity of the expectation operator we have the telescoping sum

$$\mathbb{E}[\widehat{P}_L] = \mathbb{E}[\widehat{P}_0] + \sum_{l=1}^{L} \mathbb{E}[\widehat{P}_l - \widehat{P}_{l-1}]. \tag{11}$$

In multilevel Monte Carlo, we use the expansion on the right hand side as an indirect means to evaluate the left hand side. This may be thought of as a recursive application of the *control variate* technique, which is widely used in applied statistics [23, 26, 40, 41]. To estimate $\mathbb{E}[\widehat{P}_0]$ we form the usual sample mean, based on, say, N_0, paths. This gives

$$\widehat{Y}_0 = \frac{1}{N_0} \sum_{s=1}^{N_0} \widehat{P}_0^{[s]}. \tag{12}$$

Generally, for $\mathbb{E}[\widehat{P}_l - \widehat{P}_{l-1}]$ with $l > 0$ we will use N_l paths so that

$$\widehat{Y}_l = \frac{1}{N_l} \sum_{s=1}^{N_l} \left(\widehat{P}_l^{[s]} - \widehat{P}_{l-1}^{[s]} \right). \tag{13}$$

It is vital to point out that $\widehat{P}_l^{[s]}$ and $\widehat{P}_{l-1}^{[s]}$ in (13) come from the *same discretized Brownian path*, with different stepsizes Δt_l and Δt_{l-1}, respectively. Figure 2 illustrates the idea for the case $K = 2$. In words, at a general level l, we compute N_l Brownian paths and, for each path, apply Euler–Maruyama twice; once with stepsize Δt_l and once with stepsize Δt_{l-1}. (In practice, we compute a path at resolution Δt_l and then combine Brownian increments over pairs of steps in order to get a path at resolution Δt_{l-1}.) Having constructed our N_l independent paths for level l, we start afresh at level $l + 1$; none of the earlier information is re-used and new (independent) pseudo-random numbers are generated.

Because of the choice of L in (10) we know from (11) that our estimator will have the required $O(\epsilon)$ bias. Now we will see how to choose the values of $\{N_l\}_{l=0}^{L}$ to achieve the corresponding accuracy in the overall confidence interval.

Considering a general level where $l > 0$, appealing to the strong convergence behaviour (6) of Euler–Maruyama and our assumption that h is globally Lipschitz, we have

$$
\begin{aligned}
\mathsf{var}[\widehat{P}_l - h(X(T))] &= \mathbb{E}[(\widehat{P}_l - h(X(T)))^2] - (\mathbb{E}[\widehat{P}_l - h(X(T))])^2 & (14) \\
&\leq \mathbb{E}[(\widehat{P}_l - h(X(T)))^2] & (15) \\
&\leq \text{constant} \times \mathbb{E}[(X_N - X(T))^2] & (16) \\
&= O(\Delta t_l). & (17)
\end{aligned}
$$

It then follows from Minkowski's Inequality [11] that

$$
\begin{aligned}
\mathsf{var}[\widehat{P}_l - \widehat{P}_{l-1}] &= \mathsf{var}[\widehat{P}_l - h(X(T)) + h(X(T)) - \widehat{P}_{l-1}] \\
&\leq \left(\sqrt{\mathsf{var}[\widehat{P}_l - h(X(T))]} + \sqrt{\mathsf{var}[h(X(T)) - \widehat{P}_{l-1}]} \right)^2 \\
&= O(\Delta t_l). & (18)
\end{aligned}
$$

Applying this result in (13) we conclude that $\widehat{Y}_l$ has a variance of $O(\Delta t_l/N_l)$ for $l > 1$. Because all levels are independent, we deduce that

$$
\mathsf{var}\left[\widehat{Y}_0 + \sum_{l=1}^{L} \widehat{Y}_l \right] = \mathsf{var}[\widehat{Y}_0] + \sum_{l=1}^{L} O(\Delta t_l/N_l).
$$

To balance the variance evenly across levels $l = 1, 2, \ldots, L$, and to control the variance at level $l = 0$, we choose $N_l = O(\epsilon^{-2} L \Delta t_l)$. It then follows that our overall estimator has variance

$$
O(\epsilon^2) + \sum_{l=1}^{L} O(\epsilon^2/L) = O(\epsilon^2).
$$

In this way, we have achieved the bias and variance required to give a confidence interval of the specified ϵ level of accuracy.

To quantify how the complexity of this algorithm scales with ϵ, we sum the cost of level l from $l = 0$ to L to give

$$
\sum_{l=0}^{L} N_l \Delta t_l^{-1} = \sum_{l=0}^{L} \epsilon^{-2} L \Delta t_l \Delta t_l^{-1} = L^2 \epsilon^{-2}.
$$

From (10) this expression becomes $O(\epsilon^{-2} (\log \epsilon)^2)$, as we quoted in section 3.

At this stage, a few remarks are in order:

Constructive Upper Bound: In the course of the analysis above, we came up with a general-purpose choice for the number of paths at each level, $\{N_l\}_{l=0}^{L}$. The final complexity count is therefore an upper bound on the best possible value. In practice, for a given problem and accuracy requirement, we can perform a cheap pre-processing step where appropriate variances are estimated and an optimization problem is solved in order to give a sequence $\{N_l\}_{l=0}^{L}$; see, for example, [21].

Weak versus Strong: The key inequality (18), which guarantees tight coupling between coarse and fine paths, made use of the strong convergence property. For small Δt_l, both paths are close to the true path, so the paths must be close to each other. In this sense, both strong and weak error rates are key ingredients in the analysis. We note, however, that [6] introduces a multilevel approach based on tightly coupled path simulation with schemes that are weakly, but not strongly, convergent.

Variance and Second Moment In deriving the inequality (17), we discarded the square of the first moment and used the second moment as an upper bound for the variance. This may appear to be a very crude step, but in our context it does not degrade the final conclusion. (In a different, Poisson-driven setting where a multilevel method was developed and analysed, the step (14)–(15) is no longer optimal—it is beneficial to analyse the variance directly [3].)

Exploiting Structure: As mentioned above, multilevel Monte Carlo may be viewed as a recursive version of the control variate approach. In the simplest version of control variates, if we wish to compute $\mathbb{E}[X]$, we may instead compute $\mathbb{E}[X - Y]$ and add $\mathbb{E}[Y]$, where Y is a suitably constructed random variable such that $X - Y$ has small variance and $\mathbb{E}[Y]$ is readily available [23, 26]. However, the success of this technique usually relies on incorporating some extra knowledge of the problem: a structure such as symmetry or convexity, or the existence of a "nearby" problem that is analytically tractable. In this respect, the multilevel Monte Carlo method for SDEs is very different from traditional control variates: the analysis is completely general and no special insights are needed about the nature of the underlying SDE, other than knowledge of the basic weak and strong convergence properties.

Multilevel versus Multigrid: In [19], Giles explains that the *multigrid* approach in numerical PDEs was "the inspiration for the author in developing the MLMC method for SDE path simulation." There are clear similarities between the two: the use of geometrically refined/coarsened

grids and the idea that relatively little work needs to be expended on the fine grids in order to resolve high frequency components. Indeed the multigrid idea has been applied directly to the Monte Carlo context [9]. However, it is important to keep in mind that there are also conceptual differences between the two techniques: multilevel Monte Carlo, in the sense discussed here, is distinct, and novel. For example, multilevel Monte Carlo does not involve the notion of passing information up and down the refinement levels, as is done with multigrid V or W cycles.

Related Earlier Methods: As discussed in [19, section 1.3], related earlier work on improving Monte Carlo when samples are generated via discretization was performed by Heinrich, see, for example [24, 25], and Kebaier [32] devised a two-level approach to path simulation.

Based on the type of analysis that we summarized above, it is possible to state a general theorem about multilevel simulation:

Theorem 5.1 (Giles; see for example, [19]) *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, and positive constants $\alpha, \beta, \gamma, c_1, c_2, c_3$ such that $\alpha \geq \frac{1}{2}\min(\beta,\gamma)$ and*

1. $|\mathbb{E}[P_l - P]| \leq c_1 2^{-\alpha l}$

2. $\mathbb{E}[Y_0] = \mathbb{E}[P_0]$ and $\mathbb{E}[Y_l] = \mathbb{E}[P_l - P_{l-1}]$ for $l > 0$

3. $\mathsf{var}[Y_l] \leq c_2 N_l^{-1} 2^{-\beta l}$

4. $\mathbb{E}[C_l] \leq c_3 N_l 2^{\gamma l}$, where C_l is the computational complexity of Y_l,

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

$$Y = \sum_{l=0}^{L} Y_l$$

has a mean-square error with bound

$$\mathbb{E}\left[(Y - \mathbb{E}[P])^2\right] < \epsilon^2$$

with a computational complexity C with bound

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

Giles [16] has also shown how to construct estimators for which $\beta > \gamma = 1$, by replacing Euler–Maruyama with the more strongly accurate Milstein scheme. For European-style options with Lipschitz payoff functions, this makes $O(\epsilon^{-2})$ complexity achievable. From the arguments in section (3), it is intuitively reasonable that this level of cost should be optimal in a well-defined sense. The issue is formalized in [38], where (almost) optimality of the multilevel approach is confirmed.

In section 2 we mentioned that the basic Euler–Maruyama method (1) may fail to converge in a weak or strong sense on nonlinear SDEs in the asymptotic limit $\Delta t \to 0$. A closely related question, of direct relevance to this review, is whether the combination of "Euler–Maruyama plus Monte Carlo" converges for nonlinear SDEs in the $\epsilon \to 0$ limit. In [28, 29], Hutzenthaler and Jentzen showed that Euler–Maruyama Monte Carlo can achieve convergence in a $\mathbb{P}$-almost sure sense in cases where the underlying Euler–Maruyama scheme diverges. This can happen when the events causing Euler–Maruyama to diverge are so rare that they are extremely unlikely to impact on any of the Monte Carlo samples. However, in [31] Hutzenthaler, Jentzen and Platen showed that the multilevel Monte Carlo method does not inherit this property. They established this result using a counterexample of the form

$$dX(t) = -X(t)^5 dt, \tag{19}$$

with $X(0)$ having a standard Gaussian distribution, where $\mathbb{E}[X(t)^2]$ is the required moment. Note that the SDE (19) has a zero diffusion term, so it may also be regarded as a random ODE. A modified version of Euler–Maruyama, known as a *tamed* method, was shown in [31] to recover convergence in the multilevel setting.

6 Computational Experiments

Asymptotic, $\epsilon \to 0$, analysis indicates that multilevel Monte Carlo offers a dramatic improvement in computational complexity. Numerous computational studies have confirmed that this potential can be realised in practice.

Giles has made MATLAB code available at

http://people.maths.ox.ac.uk/gilesm/acta/

that can be used as the basis for computational experimentation. In Figure 3 we show results based on this code. Here, we have an asset model given by geometric Brownian motion

$$dX(t) = 0.05 X(t) dt + 0.25 X(t) dW(t), \quad X(0) = 100.$$

We consider (a) a European call and (b) a digital call option over $[0, T]$ with $T = 1$ and exercise price 100. So the payoff functions, after discounting for interest, are

$$h(x) = e^{-0.05T} \max(x - 100, 0)$$

for the call option and

$$h(x) = \begin{cases} e^{-0.05T} 100 & \text{when } x > 100 \\ 0 & \text{when } x < 100 \end{cases}$$

for the digital option. (For those who worry about probability zero events, the code defines $h(100) = e^{-0.05T}(100+0)/2$ in the digital case.) The code repeats the Monte Carlo simulation for accuracy requests of $\epsilon = 0.1, 0.05, 0.02, 0.01, 0.005$. The upper left picture in Figure 3 shows, for the call option, the number of paths N_l used at each level l in the multilevel method. We see that for a given ϵ more paths are used at the cheaper (small l) levels, and as ϵ is decreased, so that more accuracy is required, extra levels are added. The upper right picture indicates the corresponding computational cost in terms of run time. More precisely, the asterisks (joined by dashed lines) show the cost weighted by ϵ^2 as a function of ϵ. We see that this quantity remains approximately constant, as predicted by the analysis. The picture also shows the scaled cost for an equivalent standard Monte Carlo computation, using a solid linetype. We see a much larger cost that appears to grow faster than ϵ^{-2}. The lower pictures in Figure 3 give the same results for the case of the digital option, where the payoff function is not globally Lipschitz. Again the multilevel version is seen to be more efficient than standard Monte Carlo.

7   Follow-on Research

In this section we summarize some of the key advances that have been made since the original multilevel breakthrough [17]. We focus on work that is directly relevant to financial option valuation. The comprehensive overviews [19, 21] can be consulted for further details on these, and other, areas. The webpage maintained by Giles at

`http://people.maths.ox.ac.uk/~gilesm/mlmc_community.html`

is also an excellent source of up-to-date information.

7.1   Beyond European Calls and Puts

A key step in the analysis of section 5 was to show that the coarse and refined paths are tightly coupled, in the sense that they produce payoffs

whose difference has small variance. The logic behind the analysis may be loosely summarized as

A strong convergence of Euler–Maruyama $\Rightarrow$

B coarse and refined paths close to the true path $\Rightarrow$

C coarse and refined paths close to each other $\Rightarrow$

D coarse and refined payoffs close to each other.

The C $\Rightarrow$ D step appealed to the global Lipschitz property of h. This is valid for European call and put options, where $h(x) = \max(x - E, 0)$ and $h(x) = \max(E - x, 0)$, respectively. However, the analysis must be refined for those European-style options where $\mathbb{E}[h(X(T))]$ is required for functions h that violate the global Lipschitz criterion. We may also wish to deal with *path-dependent* options where an expected value operation is applied to a functional depending on some or all of the values $X(t)$ for $0 \le t \le T$.

These more exotic options include problematic classes where, for certain SDE paths, the payoff may be very sensitive to small changes. For example, with digital options that expire close to the money, a small change in the asset path can lead to an $O(1)$ change in the payoff. Similarly, the payoff from a barrier option is very sensitive to those paths that flirt with the barrier. In these cases, the logical flow above above must be adapted. Intuitively, we should be able to exploit the fact that troublesome paths are the exception rather than the rule, and hence C $\Rightarrow$ D *with high probability*. In some cases this allows us to recover the computational complexity that we saw for European calls and puts. In other cases we must accept a slight increase in cost.

The behaviour of multilevel Monte Carlo for Asian, lookback and digital options was considered computationally in the original work of Giles [17]. Rigorous analysis to back up these results for barrier, lookback and digital options was first given in [20]. Further work has been targeted at binary options [5], Asian options [2], basket options [18], barrier options [13] and American options [8]. The use of multilevel Monte Carlo to compute sensitivities with respect to problem parameters, that is, Greeks, was considered in [10].

7.2   Further Developments

It is common practice to combine more than one variance reduction technique. Given that antithetic variables can be effective in option valuation

[23, 26], it is natural to consider embedding this approach within the multilevel framework. Giles and Szpruch [14, 22] have shown that this can be effective, particularly when Milstein is used for the numerical integration. A conditional Monte Carlo approach has also been shown to be fruitful in the mutlilevel setting [16]. In a different direction, Rhee and Glynn [39] have proposed an extra level of randomization that produces an unbiased multilevel estimator.

To go beyond $O(\epsilon^{-2})$ complexity it is possible to move to quasi Monte Carlo, where a low-discrepancy sequence replaces a pseudo-random sequence. Giles and Waterhouse [15] have demonstrated that a combination of quasi Monte Carlo and multilevel can outperform each separate technique. Belomestny and Nagapetyan [7] have also developed a multilevel Monte Carlo approach that, for appropriate classes of SDE, breaks the $O(\epsilon^{-2})$ complexity barrier.

Finally, we note that the multilevel methodology has also been extended to asset models that are not driven purely by Brownian motion [12, 43].

8   Discussion

Our aim in this article was to explain in an accessible manner the key ideas behind the multilevel Monte Carlo method. We focussed on the case of SDE-based financial option valuation, where Monte Carlo is a widely used tool. At the heart of the technique is a very general and widely applicable philosophy—a recursive application of control variates that relies on tight coupling between simulations at different resolutions. The resulting algorithm is sufficiently simple and effective that it can be implemented straightforwardly and used to produce noticable gains in computational efficiency in very general circumstances. However, as evidenced by the wealth of current research activity, there is also substantial scope for (a) refining the multilevel approach in order to exploit problem-specific information and (b) developing multilevel methods in many other stochastic simulation scenarios. For these reasons we envisage multilevel Monte Carlo evolving into a cornerstone of computational finance.

References

[1] Y. AIT-SAHALIA, *Testing continuous-time models of the spot interest rate*, Review of Financial Studies, (1999), pp. 385–426.

[2] M. B. ALAYA AND A. KEBAIER, *Multilevel Monte Carlo for Asian options and limit theorems*, Monte Carlo Methods and Applications, 20 (2014), pp. 181–194.

[3] D. F. ANDERSON, D. J. HIGHAM, AND Y. SUN, *Complexity of multilevel Monte Carlo tau-leaping*, SIAM Journal on Numerical Analysis, 52 (2015), pp. 3106–3127.

[4] D. F. ANDERSON AND J. C. MATTINGLY, *A weak trapezoidal method for a class of stochastic differential equations*, Communications in Mathematical Sciences, 9 (2011), pp. 301–318.

[5] R. AVIKAINEN, *Convergence rates for approximations of functionals of SDEs*, Finance and Stochastics, 13 (2009), pp. 381–401.

[6] D. BELOMESTNY AND T. NAGAPETYAN, *Multilevel path simulation for weak approximation schemes*, arXiv:1406.2581, (2014).

[7] ——, *Variance reduced multilevel path simulation: going beyond the complexity ϵ^{-2}*, arXiv:1412.4045, (2014).

[8] D. BELOMESTNY, J. SCHOENMAKERS, AND F. DICKMANN, *Multilevel dual approach for pricing American style derivatives*, Finance and Stochastics, 17 (2013), pp. 717–742.

[9] A. BRANDT, V. ILYIN, AND A. SKARBOVIYCHUK, *Multilevel Monte Carlo methods for studying fluids under gravity*, tech. rep., The Weizmann Institute of Science, Israel, 2003.

[10] S. BURGOS AND M. B. GILES, *Computing Greeks using multilevel path simulation*, in Monte Carlo and Quasi Monte Carlo Methods 2010, L. Plaskota and H. Wozniakowski, eds., Springer, 2012, pp. 281–296.

[11] M. CAPIŃSKI AND E. KOPP, *Measure, Integral and Probability*, Springer, Berlin, 1999.

[12] S. DEREICH AND F. HEIDENREICH, *A multilevel Monte Carlo algorithm for Lévy-driven stochastic differential equations*, Stochastic Processes and their Applications, (2011), pp. 1565–1587.

[13] M. GILES, K. DEBRABANT, AND A. RÖSSLER, *Numerical analysis of multilevel Monte Carlo path simulation using the Milstein discretisation*, ArXiv preprint: 1302.4676, (2013).

[14] M. GILES AND L. SZPRUCH, *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, 2013, pp. 367–384.

[15] M. GILES AND B. WATERHOUSE, *Multilevel quasi-Monte Carlo path simulation*, in Advanced Financial Modelling, Radon Series on Computational and Applied Mathematics, De Gruyter, 2009, pp. 165–181.

[16] M. B. GILES, *Improved multilevel Monte Carlo convergence using the Milstein scheme*, in Monte Carlo and Quasi-Monte Carlo Methods, Springer, 2007, pp. 343–358.

[17] ——, *Multilevel Monte Carlo path simulation*, Operations Research, 56 (2008), pp. 607–617.

[18] ——, *Multilevel Monte Carlo for basket options*, in Proceedings of the 2009 Winter Simulation Conference, Austin, M. D. Rossetti, R. R. Hill, B. Johansson, A. Dunkin, and R. G. Ingalls, eds., IEEE, 2009, pp. 1283–1290.

[19] ——, *Multilevel Monte Carlo methods*, in Monte Carlo and Quasi Monte Carlo Methods 2012, J. Dick, F. Y. Kuo, G. W. Peters, and I. H. Sloan, eds., Springer, 2014, pp. 79–98.

[20] M. B. GILES, D. J. HIGHAM, AND X. MAO, *Analysing multi-level Monte Carlo for options with non-globally Lipschitz payoff*, Finance and Stochastics, 13 (2009), pp. 403–413.

[21] M. B. GILES AND L. SZPRUCH, *Multilevel Monte Carlo methods for applications in finance*, in Recent Developments in Computational Finance, T. Gerstner and P. E. Kloeden, eds., World Scientific, 2013.

[22] ——, *Antithetic multilevel Monte Carlo estimation for multi-dimensional SDEs without Lévy area simulation*, Annals of Applied Probability, 24 (2014), pp. 15850–1620.

[23] P. GLASSERMAN, *Monte Carlo Methods in Financial Engineering*, Springer, Berlin, 2004.

[24] S. Heinrich, *Monte Carlo complexity of global solution of integral equations*, Journal of Complexity, 14 (1998), pp. 151–175.

[25] ——, *Monte Carlo approximation of weakly singular integral operators*, Journal of Complexity, 22 (2006), pp. 192–219.

[26] D. J. Higham, *An Introduction to Financial Option Valuation: Mathematics, Stochastics and Computation*, Cambridge University Press, Cambridge, 2004.

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

[28] M. Hutzenthaler and A. Jentzen, *Convergence of the stochastic Euler scheme for locally Lipschitz coefficients*, Found. Comput. Math., 11 (2011), pp. 657–706.

[29] ——, *Numerical approximations of stochastic differential equations with non-globally Lipschitz continuous coefficients*, Memoirs of the American Mathematical Society, 236 (2014), p. in press.

[30] M. Hutzenthaler, A. Jentzen, and P. E. Kloeden, *Strong and weak divergence in finite time of Euler's method for stochastic differential equations with non-globally Lipschitz continuous coefficients*, Proc. R. Soc. A, 467 (2011), pp. 1563–1576.

[31] M. Hutzenthaler, A. Jentzen, and P. E. Kloeden, *Divergence of the multilevel Monte Carlo Euler method for nonlinear stochastic differential equations*, Annals of Applied Probability, 23 (2013), pp. 1913–1966.

[32] A. Kebaier, *Statistical Romberg extrapolation: a new variance reduction method and applications to option pricing*, Annals of Applied Probabilty, 14 (2005), pp. 2681–2705.

[33] P. E. Kloeden and A. Neuenkirch, *The pathwise convergence of approximation schemes for stochastic differential equations*, London Mathematical Society Journal of Computation and Mathematics, 10 (2007), pp. 235–253.

[34] P. E. Kloeden and E. Platen, *Numerical Solution of Stochastic Differential Equations*, Springer Verlag, Berlin, Third Printing, 1999.

[35] T. Mikosch, *Elementary Stochastic Calculus (with Finance in View)*, World Scientific, Singapore, 1998.

[36] G. N. Milstein and M. V. Tretyakov, *Stochastic Numerics for Mathematical Physics*, Springer-Verlag, Berlin, 2004.

[37] G. N. Milstein and M. V. Tretyakov, *Numerical integration of stochastic differential equations with nonglobally Lipschitz coefficients*, SIAM Journal on Numerical Analysis, 43 (2005), pp. 1139–1154.

[38] T. Müller-Gronbach and K. Ritter, *Variable subspace sampling and multi-level algorithms*, in Monte Carlo and Quasi-Monte Carlo Methods 2008, P. L'Ecuyer and A. Owen, eds., Springer, 2009, pp. 131–156.

[39] C. Rhee and P. W. Glynn, *A new approach to unbiased estimation for SDEs*, in Proceedings of the 2012 Winter Simulation Conference, C. Laroque, J. Himmelspach, R. Pasupathy, O. Rose, and A. Uhrmacher, eds., 2012, pp. 201–207.

[40] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*, Springer, Berlin, 2nd ed., 2004.

[41] R. Seydel, *Tools for Computational Finance*, Springer, Berlin, fifth ed., 2012.

[42] L. Szpruch, X. Mao, D. J. Higham, and J. Pan, *Strongly nonlinear Ait-Sahalia-type interest rate model and its numerical approximation*, BIT Numerical Mathematics, 51 (2011), pp. 405–425.

[43] Y. Xia and M. B. Giles, *Multilevel path simulation for jump-diffusion SDEs*, in Monte Carlo and Quasi Monte Carlo Methods 2010, L. Plaskota and H. Wozniakowski, eds., Springer, 2012, pp. 695–708.
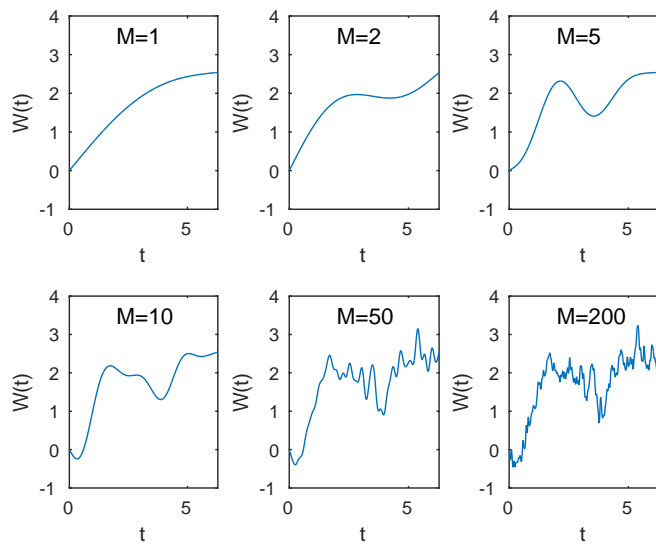
Figure 1: Paths based on the Paley-Wiener representation (8). As indicated, the six plots show the sum truncated after $M = 1, 2, 5, 10, 50$ and 200 sine terms.
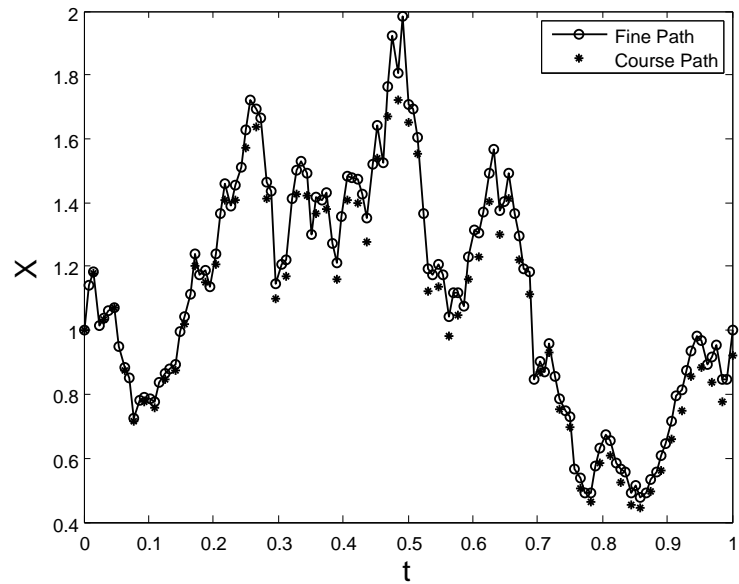
Figure 2: Illustration of how the estimator $\widehat{Y}_l$ in (13) is constructed. Circles (joined by straight lines for clarity) show the refined Euler–Maruyama path, with stepsize $\Delta t = 2^{-l}T$. Asterisks show the coarser Euler–Maruyama path, with stepsize $\Delta t = 2^{-l+1}T$, *computed with the same Brownian increments.*
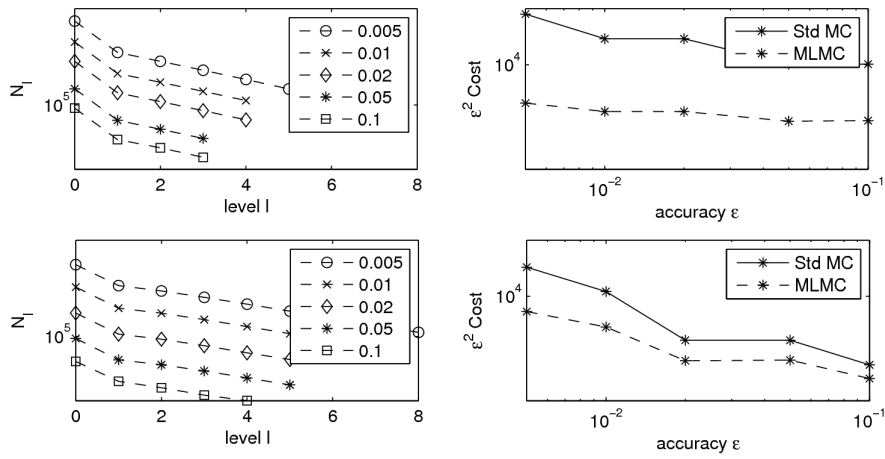
Figure 3: Output from the multilevel Monte Carlo code made available by Giles (see text for web site address). Left hand pictures show the number of paths per level at each target accuracy. Right hand pictures show the computation time, scaled by ϵ^2. Upper pictures are for a European call option. Lower pictures are for a digital option.