Improving MLMC for SDEs with application to the Langevin equation

Eike H. Müller   Rob Scheichl   Tony Shardlow

December 23, 2014

Abstract

This paper applies several well-known tricks from the numerical treatment of deterministic differential equations to improve the efficiency of the Multilevel Monte Carlo (MLMC) method for stochastic differential equations (SDEs) and especially the Langevin equation. We use modified equations analysis to circumvent the need for a strong-approximation theory for the integrator, and we apply this to introduce MLMC for Langevin-type equations with integrators based on operator splitting. We combine this with extrapolation and investigate the use of discrete random variables in place of the Gaussian increments, which is a well-known technique for the weak approximation of SDEs. We show that, for small-noise problems, discrete random variables can lead to an increase in efficiency of almost two orders of magnitude for practical levels of accuracy.

Keywords numerical solution of stochastic differential equations, modified equations, geometric integrators, weak approximation, extrapolation.

1   Introduction

This paper is concerned with the numerical solution of stochastic differential equations (SDEs) by the Multilevel Monte Carlo (MLMC) method. MLMC (Giles, 2008; Heinrich, 2001) is an important variance-reduction method that is well established by now and has been successfully applied to a wide class of problems in stochastic simulation and in uncertainty quantification; for example, (Anderson & Higham, 2011; Barth *et al.*, 2011; Cliffe *et al.*, 2011; Dereich & Heidenreich, 2011; Giles & Reisinger, 2012; Giles & Szpruch, 2013; Hoel *et al.*, 2012; Mishra *et al.*, 2012). The variance reduction in MLMC is achieved by computing approximations of the solution on different "levels" consisting, in the SDE case, of numerical integrators with different time-step sizes. These computations are then combined in an efficient way to define a multilevel estimator for the moments that has a smaller variance than the standard Monte Carlo estimator and can therefore be computed faster.

Let $(\Omega, \mathcal{F}, \mathbb{P})$ denote a probability space and let $\mathbb{E}$ and Var denote the expectation and variance with respect to $\mathbb{P}$. Consider first the initial-value problem

$$d\boldsymbol{X} = \boldsymbol{f}(\boldsymbol{X})\, dt + G(\boldsymbol{X})\, d\boldsymbol{W}(t), \qquad \boldsymbol{X}(0) = \boldsymbol{X}_0, \tag{1.1}$$

for $\boldsymbol{f} \colon \mathbb{R}^d \to \mathbb{R}^d$ and $G \colon \mathbb{R}^d \to \mathbb{R}^{d \times m}$ and initial data $\boldsymbol{X}_0 \in \mathbb{R}^d$. Here $\boldsymbol{W}(t)$ is a vector of m *iid* standard Brownian motions on $(\Omega, \mathcal{F}, \mathbb{P})$. Suppose that there exists a well-defined solution $\boldsymbol{X}(t)$ when Eq. (1.1) is interpreted as an Ito integral equation. For simplicity, we only consider approximating moments of the solution at a prescribed end time as the quantities of interest, but other, more complicated functionals could also be studied. That is, we are interested in computing $\mathbb{E}[\phi(\boldsymbol{X}(T))]$ for some $\phi \colon \mathbb{R}^d \to \mathbb{R}$ and time $T > 0$. Consider the approximation by a sequence of random variables $\boldsymbol{X}_n \approx \boldsymbol{X}(t_n)$ for $t_n = nh$ with $n \in \mathbb{N}$ and a time step h. For example, $\boldsymbol{X}_n$ may result from the Euler–Maruyama method

$$\boldsymbol{X}_{n+1} = \boldsymbol{X}_n + \boldsymbol{f}(\boldsymbol{X}_n)h + G(\boldsymbol{X}_n)\sqrt{h}\,\boldsymbol{\xi}_n, \tag{1.2}$$

with $\boldsymbol{\xi}_n \sim \mathrm{N}(0, I)$ *iid*. In this case, $\boldsymbol{X}_n$ is a weak first-order approximation to $\boldsymbol{X}(t_n)$ so that, for any $\phi \colon \mathbb{R}^d \to \mathbb{R}$ in a suitable class of test functions $\mathcal{C}$,

$$\sup_{0 \le t_n \le T} \mathbb{E}\big[\phi(\boldsymbol{X}(t_n))\big] - \mathbb{E}\big[\phi(\boldsymbol{X}_n)\big] = \mathcal{O}\big(h\big).$$

If $\boldsymbol{f}$ and G are sufficiently smooth, $\mathcal{C}$ contains all infinitely differentiable functions whose derivatives are polynomially bounded; for example, (Kloeden & Platen, 1992, Theorem 14.5.1).

In some cases (Shardlow, 2006; Zygalakis, 2011), it is possible to find a second SDE, called the modified SDE with solution $\boldsymbol{X}_h(t)$, such that $\boldsymbol{X}_n$ is a second-order weak approximation to $\boldsymbol{X}_h(t)$; that is,

$$\sup_{0 \le t_n \le T} \mathbb{E}\big[\phi(\boldsymbol{X}_h(t_n))\big] - \mathbb{E}\big[\phi(\boldsymbol{X}_n)\big] = \mathcal{O}\big(h^2\big). \tag{1.3}$$

Then, the solution of the modified equation $\boldsymbol{X}_h(t)$ is an order of h closer to the numerical solution than $\boldsymbol{X}(t)$. The modified equation takes the form

$$d\boldsymbol{X}_h = \tilde{\boldsymbol{f}}(\boldsymbol{X}_h)\,dt + \widetilde{G}(\boldsymbol{X}_h)\,d\boldsymbol{W}(t), \qquad \boldsymbol{X}(0) = \boldsymbol{X}_0, \tag{1.4}$$

where $\tilde{\boldsymbol{f}} = \boldsymbol{f} + h\boldsymbol{f}_1$ and $\widetilde{G} = G + hG_1$ for some $\boldsymbol{f}_1 \colon \mathbb{R}^d \to \mathbb{R}^d$ and $G_1 \colon \mathbb{R}^d \to \mathbb{R}^{d \times m}$. This reduces to Eq. (1.1) with $h = 0$, and $\boldsymbol{f}_1$ and G_1 describe the correction in the drift and diffusion needed to achieve Eq. (1.3). Our results concern SDEs and numerical integrators where the second-order modified equation is available. Except in special cases (e.g., if G is independent of $\boldsymbol{X}$), this does not include the Euler–Maruyama method (Shardlow, 2006). It does include the Milstein method, which has a second-order modified equation (Zygalakis, 2011). Using weak-approximation theory and modified equations, we develop an alternative method of analysis for MLMC in this paper. By doing this, we no longer depend directly on the strong-approximation properties of the integrator (as in other papers, e.g. (Giles, 2008)) and this gives greater freedom in the application of MLMC.

We focus on a class of integrators for an important model in molecular dynamics and atmospheric dispersion, the *Langevin equation*:

$$\begin{aligned} d\boldsymbol{P} &= -\lambda \boldsymbol{P}\,dt - \nabla V(\boldsymbol{Q})\,dt + \sigma\,d\boldsymbol{W}(t), \\ d\boldsymbol{Q} &= \boldsymbol{P}\,dt \end{aligned} \tag{1.5}$$

for parameters $\lambda, \sigma > 0$, a potential $V \colon \mathbb{R}^d \to \mathbb{R}$, and a d-dimensional vector $\boldsymbol{W}(t)$ of *iid* Brownian motions. We specify initial conditions $(\boldsymbol{Q}(0), \boldsymbol{P}(0)) = (\boldsymbol{Q}_0, \boldsymbol{P}_0) \in \mathbb{R}^{2d}$. This system is used in molecular dynamics to simulate a system of particles in a heat bath and has equilibrium distribution with pdf $Z^{-1} \exp(-H(\boldsymbol{Q}, \boldsymbol{P})/k_{\mathrm{B}}T)$, known as the *Gibbs canonical distribution*, where Z is a normalisation constant, $H(\boldsymbol{Q}, \boldsymbol{P}) \coloneqq \frac{1}{2}\boldsymbol{P}^{\mathsf{T}}\boldsymbol{P} + V(\boldsymbol{Q})$, and $k_{\mathrm{B}}T = \sigma^2/2\lambda$. As usual, k_{B} denotes the Boltzmann constant and T temperature. The Langevin equation is also used to model the dispersion of atmospheric pollutants in homogeneous turbulence (Rodean, 1996). In that case, λ is the inverse velocity autocorrelation time and $\sqrt{\sigma^2/2\lambda}$ is the strength of turbulent velocity fluctuations, and d is equal to the number of space dimensions. This is much smaller than in molecular dynamics applications, where d is proportional to the number of particles. With a slight generalisation, it can also be used to model the dispersion in inhomogeneous turbulence.

Numerical integrators for the Langevin equation are well developed for example in (Beard & Schlick, 2000; Brunge *et al.*, 1984; Wang & Skeel, 2003). Recently, there has been a strong push to understand the invariant measure associated to the integrators (Abdulle *et al.*, 2014; Bou-Rabee & Owhadi, 2010; Debussche & Faou, 2012; Kopec, 2013; Leimkuhler *et al.*, 2013, 2014; Leimkuhler & Matthews, 2013; Zygalakis, 2011). Second-order modified equations are available for the most important integrators for the Langevin equation. In particular, we study splitting methods based on exact sampling of an Ornstein–Uhlenbeck process and symplectic integrators (symplectic Euler and Störmer–Verlet) for the Hamiltonian part. We show how to couple the different levels and apply MLMC with these methods. We find the use of the exact Ornstein–Uhlenbeck process is particularly effective when λ is large.

We also combine these new integrators with extrapolation (Talay & Tubaro, 1990). It is a natural addition to MLMC methods, already mentioned in the original work (Giles, 2008) and studied in more detail in (Lemaire & Pagès, 2013). It reduces the bias in the numerical approximation of the solution due to time stepping and relies on having a sharp estimate for the bias error. If such an estimate is available, it is possible to eliminate the leading-order error term in the bias error by extrapolating from a sequence of approximations with differing time-step sizes. These approximations are naturally available in MLMC.

We provide a set of experiments for the Langevin equation with a harmonic and a double-well potential, comparing integrators based on splitting methods and extrapolation within MLMC. Our results confirm that the splitting methods are significantly more effective than the Euler–Maruyama method when combined with MLMC. All methods have the same asymptotic ϵ-cost; that is, the cost always grows inverse proportionally to the mean-square error, but the proportionality constant is reduced by an order of magnitude from the standard Euler–Maruyama method through our enhancements.

Finally, we show how discrete random variables, as an approximation to the Gaussian increments of a Brownian motion, can be used within MLMC. This would be difficult to analyse by the standard analysis, since all the approximation results for integrators based on discrete random variables are in distribution only (e.g., (Kloeden & Platen, 1992, §14.2)). In general, one must be careful in using discrete random variables in place of Gaussian random variables. The discrete approximations do not share the property of Gaussian random variables that the sum of two independent increments is itself an increment from the same distribution and hence the telescoping sum property, which is key to the standard MLMC idea, no longer holds. However, for a practical range of parameters in small-noise problems, the extra bias introduced is small and easy to estimate. Accepting this extra bias can lead to a significant improvement in efficiency, since discrete random variables allow the exact evaluation of the expected value on the coarsest level. The cost of this direct evaluation grows exponentially with the number of time steps, but it requires no sampling and, for a small number of time steps, its cost is significantly smaller than that of a Monte Carlo estimator. To analyse this method, we prove a new complexity theorem that allows for extra bias to be introduced between levels in MLMC.

The paper is organised as follows. §2 reviews MLMC, including the important complexity theorem. §3 uses modified equations to apply the complexity theorem, depending only on weak convergence of the integrators. §4 reviews splitting methods for the Langevin equation and defines a number of integrators where modified equations are available. Numerical experiments are presented in §5 to demonstrate the effectiveness of this methodology for the Langevin equation and to give quantitative predictions of the possible gains. A final section considers approximation of the Gaussian increments by discrete random variables and highlights the potential gains this can bring. The C++ source code that we developed for the numerical experiments is freely available for download under the LGPL 3 license.

2 Background on MLMC

When solving an SDE numerically, the total error consists of the bias due to the time-stepping method and the Monte Carlo sampling error. The sum of these two terms should be reduced below a given small tolerance ϵ. A standard Monte Carlo method achieves this by computing N sample paths, with $N^{-1} = \mathcal{O}(\epsilon^2)$, and taking time step $h = \mathcal{O}(\epsilon^{1/\alpha})$, where α is the order of weak convergence (e.g., $\alpha = 1$ for the Euler–Maruyama method). Hence, we can achieve accuracy ϵ with total cost $\mathrm{Cost}^{(\mathrm{MC})}(\epsilon) = \mathcal{O}(h^{-1} \times N) = \mathcal{O}\big(\epsilon^{-(2+1/\alpha)}\big)$. In contrast, MLMC uses a series of coarse levels with larger time steps to construct an estimator. If the strong order of convergence of the employed integrator is one, MLMC reduces the cost of the method to $\mathrm{Cost}^{(\mathrm{MLMC})}(\epsilon) = \mathcal{O}(N) = \mathcal{O}(\epsilon^{-2})$, which is the lower limit for a Monte Carlo method.

While MLMC is more efficient than standard Monte Carlo in the limit $\epsilon \to 0$, the actual value of the tolerance ϵ might be relatively large in practical applications. Hence, not only the asymptotic rate of convergence, but also the cost of the method for a given ϵ is of interest. The exact value of the constant C_2 in the cost function $\mathrm{Cost}^{(\mathrm{MLMC})}(\epsilon) = C_2 \epsilon^{-2} + \cdots$ and the size

of higher-order corrections depends on the details of the method, such as the time-stepping scheme and the coarse-level solver. In particular, choosing a time-stepping scheme that becomes unstable on the coarser levels can severely limit the performance as only a small number of levels can be used; see (Abdulle & Blumenthal, 2013; Hutzenthaler *et al.*, 2013).

Suppose that we are interested in the expectation of $\mathcal{P} := \phi(\boldsymbol{X}(T))$, where $\boldsymbol{X}(T)$ is the solution to Eq. (1.1) at time T and $\phi \colon \mathbb{R}^d \to \mathbb{R}$ defines the quantity of interest. Assume that the number of time steps used to discretise the SDE is $M = M_0 2^L$, where $M_0, L \in \mathbb{N}$. Our strategy is to approximate Eq. (1.1) using a numerical integrator with time step $h = T/M$ to define an approximate solution $\boldsymbol{X}_M \approx \boldsymbol{X}(T)$. Then, we compute many independent samples of $\boldsymbol{X}_M$ to define approximate samples $\mathcal{P}^{(i)}$ of $\mathcal{P}$. The classical Monte Carlo method approximates $\mathbb{E}[\mathcal{P}]$ by the sample average of $\mathcal{P}^{(i)}$.

Instead, the MLMC method constructs a sequence of approximations on levels indexed by $\ell = \{L, L-1, \ldots, 0\}$ with $M_\ell = M_0 2^\ell$ time steps of size $h_\ell = T/M_\ell$. Let $\mathcal{P}_\ell^{(i)}$ denote independent samples of the approximation to $\mathcal{P}$ on level ℓ and let

$$\widehat{\mathcal{P}}_\ell := \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} \mathcal{P}_\ell^{(i)} \tag{2.1}$$

denote the Monte Carlo estimator on level ℓ based on N_ℓ samples. An estimator for the finest level where $M = M_L$ can be written as the telescoping sum

$$\widehat{\mathcal{P}}^{(\mathrm{MLMC})} \equiv \widehat{Y}_{\{N_\ell\}} := \sum_{\ell=0}^{L} \widehat{Y}_{\ell, N_\ell} \,, \tag{2.2}$$

where $\widehat{Y}_{0, N_0} := \widehat{\mathcal{P}}_0$ and

$$\widehat{Y}_{\ell, N_\ell} := \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} Y_\ell^{(i)}, \qquad Y_\ell^{(i)} := \mathcal{P}_\ell^{(i)} - \mathcal{P}_{\ell-1}^{(i)}, \quad \text{for } \ell \geq 1. \tag{2.3}$$

The estimator does not introduce any additional bias, as we recover the numerical discretisation error on the finest level (where $h = h_L$):

$$\mathbb{E}\left[\widehat{\mathcal{P}}^{(\mathrm{MLMC})}\right] = \mathbb{E}\left[\widehat{Y}_{\{N_\ell\}}\right] = \mathbb{E}\left[\widehat{\mathcal{P}}_L\right] = \mathbb{E}\left[\widehat{\mathcal{P}}^{(\mathrm{MC})}\right], \tag{2.4}$$

where $\widehat{\mathcal{P}}^{(\mathrm{MC})}$ is the standard Monte Carlo estimator for $M = M_L$ time steps. The two key ideas of the MLMC method are now:

- The number of time steps M_ℓ is smaller on the coarser levels $\ell < L$. Hence, the calculation of a single sample $\mathcal{P}_\ell^{(i)}$ is substantially cheaper.

- The success of the method depends on coupling the samples $\mathcal{P}_\ell^{(i)}$ and $\mathcal{P}_{\ell-1}^{(i)}$ so that the variance of $Y_\ell^{(i)} = \mathcal{P}_\ell^{(i)} - \mathcal{P}_{\ell-1}^{(i)}$ is small. By arranging for the variance of $Y_\ell^{(i)}$ to be small, a smaller number N_ℓ of samples suffices to construct an accurate estimator $\widehat{Y}_{\ell, N_\ell}$. This allows the construction of a MLMC estimator with fixed total variance $\sum_{\ell=0}^{L} \mathrm{Var}\left[\widehat{Y}_\ell\right]/N_\ell$ and lower computational cost.

This is formalised in the following complexity theorem (Giles, 2008, Theorem 3.1):

Theorem 2.1 (MLMC complexity)**.** *Consider a real-valued random variable $\mathcal{P}$ and estimators $\widehat{\mathcal{P}}_\ell$ corresponding to a numerical approximation based on time step $h_\ell = T/M_\ell$ and N_ℓ samples. If there exist independent estimators $\widehat{Y}_{\ell, N_\ell}$ based on N_ℓ Monte Carlo samples, and positive constants $\alpha \geq \frac{1}{2}$, c_1, c_2, c_3 such that*

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

(ii) $\quad \mathbb{E}\left[\widehat{Y}_{\ell, N_\ell}\right] = \begin{cases} \mathbb{E}\left[\widehat{\mathcal{P}}_\ell\right], & \ell = 0, \\ \mathbb{E}\left[\widehat{\mathcal{P}}_\ell - \widehat{\mathcal{P}}_{\ell-1}\right], & \ell > 0, \end{cases}$

(iii) $\quad \mathrm{Var}\left[\widehat{Y}_{\ell, N_\ell}\right] \leq c_2 N_\ell^{-1} h_\ell^2$, and

(iv) $\quad \mathrm{Cost}_\ell^{(\mathrm{MLMC})}$, the computational complexity of $\widehat{Y}_{\ell, N_\ell}$, is bounded by $c_3 N_\ell h_\ell^{-1}$,

then there exists a positive constant c_4 such that for any $\epsilon < 1/e$, there are values L and N_ℓ for which $\widehat{Y}_{\{N_\ell\}}$ from Eq. (2.2) has a mean-square error (MSE) with bound

$$MSE \equiv \mathbb{E}\left[\left(\widehat{Y}_{\{N_\ell\}} - \mathbb{E}\left[\mathcal{P}\right]\right)^2\right] < \epsilon^2 \tag{2.5}$$

and a computational complexity $\mathrm{Cost}^{(\mathrm{MLMC})}$ with bound

$$\mathrm{Cost}^{(\mathrm{MLMC})} \leq c_4 \epsilon^{-2}. \tag{2.6}$$

The theorem can be extended to allow the variance to decay as $\mathrm{Var}\,\widehat{Y}_\ell \leq c_2 N_\ell^{-1} h_\ell^\beta$ (Giles, 2008). For all cases in this paper, $\beta = 2$ and the cost is concentrated on the coarsest level (as we see from Algorithm 1 line 12 and (iii) above). The asymptotic dependence of the computational complexity on ϵ is independent of the weak order of convergence α of the time-stepping method. However, the constant c_4 does depend on the particular time-stepping method.

To obtain the results in this paper, we used Algorithm 1 and our choices for the numbers of samples N_ℓ on each of the levels are defined adaptively via N_ℓ^+ using the sample variance following (Giles, 2008). Given a tolerance $\epsilon_{\max} > 0$, the algorithm gives an MLMC estimator $\widehat{\mathcal{P}}^{(\mathrm{MLMC})}$ with mean-square error ϵ in the range $\epsilon_{\max}/2 < \epsilon < \epsilon_{\max}$ as defined in Eq. (2.5).

3 Applying the complexity theorem

Our goal is to apply the complexity theorem to numerical integrators using only weak-approximation properties of the numerical methods. The complexity theorem makes assumptions on (i) the bias, (ii) the consistency of the estimators, and (iii) the variance of the corrections. (i) can be understood from existing weak-convergence analysis. Let $C_{\mathrm{poly}}^\infty(\mathbb{R}^d)$ be the set of infinitely differentiable functions $\mathbb{R}^d \to \mathbb{R}$ such that all derivatives are polynomially bounded.

Definition 3.1. For a time step $h > 0$, let $\boldsymbol{X}_n$ be a $\mathbb{R}^d$-valued random variable that approximates the solution $\boldsymbol{X}(t)$ to Eq. (1.1) at time $t = nh$. We say $\boldsymbol{X}_n$ is a *weak order-α* approximation if, for all $\phi \in C_{\mathrm{poly}}^\infty(\mathbb{R}^d)$ and $T > 0$, there exists $K > 0$ such that for h sufficiently small

$$\left|\mathbb{E}\left[\phi(\boldsymbol{X}(T))\right] - \mathbb{E}\left[\phi(\boldsymbol{X}_M)\right]\right| \leq Kh^\alpha, \qquad Mh = T.$$

There are many integrators that provide weak order-α approximations for $\alpha = 1$ or $\alpha = 2$ (e.g., Kloeden & Platen (1992) or §4). In the case that $\mathcal{P} = \phi(\boldsymbol{X}(T))$, $T = Mh_\ell$, and $\widehat{\mathcal{P}}_\ell = \phi(\boldsymbol{X}_M)$ for an $\boldsymbol{X}_M$ with step h_ℓ that is weak α-order, the bias condition (i) holds.

The consistency of the estimators (ii) is an easy consequence of the linearity of integration and Eq. (2.3).

Condition (iii) on the variance of corrections normally follows from the mean-square convergence of the integrator (Giles, 2008). Mean-square convergence measures the approximation of individual sample paths of the solution $\boldsymbol{X}(t)$ and hence is a tool for understand the coupling of successive levels. In this paper, we use an alternative method based on weak-approximation theory and derive condition (iii) as a consequence of the existence of a second-order modified equation. To do this, we introduce the following doubled-up system for $\boldsymbol{Z} = [\boldsymbol{X}, \boldsymbol{Y}] \in \mathbb{R}^{2d}$:

$$\begin{aligned} d\boldsymbol{X} &= \boldsymbol{f}(\boldsymbol{X})\,dt + G(\boldsymbol{X})\,d\boldsymbol{W}(t), & \boldsymbol{X}(0) &= \boldsymbol{X}_0 \in \mathbb{R}^d, \\ d\boldsymbol{Y} &= \boldsymbol{f}(\boldsymbol{Y})\,dt + G(\boldsymbol{Y})\,d\boldsymbol{W}(t), & \boldsymbol{Y}(0) &= \boldsymbol{X}_0. \end{aligned} \tag{3.1}$$

Algorithm 1 Multilevel Monte Carlo. Input: $\epsilon_{\max}$, M_0, and T. Output: Estimator $\widehat{\mathcal{P}}^{(\mathrm{MLMC})}$

1: Choose L, M_0 such that, on the finest level with $M_L = 2^L M_0$ time steps of size $h_L = T/M_L$, the bias ϵ_{bias} is smaller than $\epsilon_{\max}/\sqrt{2}$. Define $\epsilon \equiv \sqrt{2}\epsilon_{\mathrm{bias}}$.

2: Choose a minimum number of samples $N_{\min}$ (say 100 or 1000).

3: Set $N_\ell^- = 1$, $N_\ell^+ = N_{\min}$, $N_\ell = 0$ for all levels ℓ.

4: **while** $N_\ell < N_\ell^+$ for some level ℓ **do**

5:    **for** $\ell = L, \ldots, 0$ **do**

6:       **for** $i = N_\ell^-, \ldots, N_\ell^+$ **do**

7:          Calculate $Y_\ell^{(i)}$ by applying the numerical integrator on levels ℓ and $\ell - 1$ (except for $\ell = 0$) for sample i. The two trajectories should be coupled (see §4.3), but $Y_\ell^{(i)}$ should be independent of any other sample (i.e., of $Y_{\ell'}^{(i')}$ for $\ell' \neq \ell$ or $i' \neq i$).

8:          $N_\ell \mapsto N_\ell + 1$.

9:       **end for**

10:       Update estimators for the bias and variance:

$$\widehat{Y}_{\ell,N_\ell} = \frac{1}{N_\ell}\sum_{i=1}^{N_\ell} Y_\ell^{(i)}, \quad \widehat{V}_{\ell,N_\ell} = \frac{1}{N_\ell - 1}\left[\sum_{i=1}^{N_\ell}\left(Y_\ell^{(i)}\right)^2 - \frac{1}{N_\ell}\left(\sum_{i=1}^{N_\ell} Y_\ell^{(i)}\right)^2\right].$$

11:       $N_\ell^- = N_\ell^+ + 1$.

12:       Calculate the optimal N_ℓ^+ according to formula (12) in (Giles, 2008):

$$N_\ell^+ = \left\lceil 2\epsilon^{-2}\sqrt{\widehat{V}_{\ell,N_\ell} h_\ell}\left(\sum_{j=0}^{L}\sqrt{\widehat{V}_{j,N_j}/h_j}\right)\right\rceil.$$

13:    **end for**

14: **end while**

15: Return estimator $\widehat{\mathcal{P}}^{(\mathrm{MLMC})} = \widehat{Y}_{\{N_\ell\}} \equiv \sum_{\ell=0}^{L}\widehat{Y}_{\ell,N_\ell}$

The same initial data is applied and the same $\boldsymbol{W}(t)$ drives both components and so $\boldsymbol{X}(t) = \boldsymbol{Y}(t)$ a.s. for $t > 0$. We now have two copies of $\boldsymbol{X}(t)$ and we approximate each differently. Formally, we approximate $\boldsymbol{X}(t)$ and $\boldsymbol{Y}(t)$ by different numerical integrators with step $h > 0$ and denote the resulting approximation to $\boldsymbol{Z}(t_n)$ by $\boldsymbol{Z}_n = [\boldsymbol{X}_n, \boldsymbol{Y}_n]$ at $t_n = nh$. In MLMC, there is usually one integrator applied with time steps h for $\boldsymbol{X}$ and $h/2$ for $\boldsymbol{Y}$ (which is a little awkward for $\boldsymbol{Y}_n$, as one increment of n corresponds to two steps of the underlying integrator). The joint distribution of the $\boldsymbol{X}_n$ and $\boldsymbol{Y}_n$ contains all the required information about the coupling of the approximations of each component and, as we now show, a weak-convergence analysis of the system gives condition (ii).

For simplicity, we start by assuming that $\boldsymbol{Z}_n = [\boldsymbol{X}_n, \boldsymbol{Y}_n]$ is a weak second-order approximation to $\boldsymbol{Z}(t) = [\boldsymbol{X}(t), \boldsymbol{Y}(t)]$. Then, we can prove the following.

Theorem 3.2. *Fix $T > 0$ and let $\mathcal{P} = \phi(\boldsymbol{X}(T))$ for a $\phi \in C_{\mathrm{poly}}^\infty(\mathbb{R}^d)$. Suppose that $\boldsymbol{Z}_n$ is a weak second-order approximation to $\boldsymbol{Z}(t)$. Conditions (i)–(iii) of Theorem 2.1 hold with $[\mathcal{P}_\ell^{(i)}, \mathcal{P}_{\ell-1}^{(i)}]$ given by* iid *samples of $[\phi(\boldsymbol{X}_M), \phi(\boldsymbol{Y}_M)]$ with $h = h_\ell$ and $Mh = T$.*

Proof. The condition on $\boldsymbol{Z}_n$ implies also that $\boldsymbol{X}_n$ and $\boldsymbol{Y}_n$ are weak second-order approximations to $\boldsymbol{X}(t)$. Then, by the above discussion, conditions (i) with $\alpha = 2$ and (ii) hold.

Let $\psi(\boldsymbol{Z}) := \phi(\boldsymbol{X}) - \phi(\boldsymbol{Y})$. Then $\psi^2 \in C_{\mathrm{poly}}^\infty(\mathbb{R}^{2d})$ since ϕ and hence ψ^2 are smooth and their derivatives are polynomially bounded. As $\boldsymbol{Z}_n$ is a weak second-order approximation to $\boldsymbol{Z}(t)$,

$$\mathbb{E}\big[\psi(\boldsymbol{Z}_M)^2 - \psi(\boldsymbol{Z}(T))^2\big] = \mathcal{O}\big(h^2\big).$$

By definition of ψ,

$$\mathbb{E}\Big[\big|\phi(\boldsymbol{X}_M) - \phi(\boldsymbol{Y}_M)\big|^2 - \big|\phi(\boldsymbol{X}(T)) - \phi(\boldsymbol{Y}(T))\big|^2\Big] = \mathcal{O}\big(h^2\big). \tag{3.2}$$

Using the fact that $\boldsymbol{X}(t) = \boldsymbol{Y}(t)$ a.s., we have

$$\mathbb{E}\Big[\big|\phi(\boldsymbol{X}_M) - \phi(\boldsymbol{Y}_M)\big|^2\Big] = \mathcal{O}\big(h^2\big).$$

Written in terms of $\mathcal{P}_\ell$ and $\mathcal{P}_{\ell-1}$, this means

$$\mathrm{Var}\big[\mathcal{P}_\ell - \mathcal{P}_{\ell-1}\big] \leq \mathbb{E}\Big[\big(\mathcal{P}_\ell - \mathcal{P}_{\ell-1}\big)^2\Big] = \mathcal{O}\big(h^2\big).$$

In other words, the variance of each sample of the coarse–fine correction is order h^2. This implies that the sample average $\widehat{Y}_{\ell, N_\ell}$ of N_ℓ *iid* samples satisfies condition (iii) of Theorem 2.1. $\quad\square$

3.1 Modified equations

The above argument does not apply to weak first-order accurate methods, even though the complexity theorem only requires $\alpha > 1/2$. In this case, we use the theory of modified equations to extend the analysis. A modified equation is a small perturbation of the original SDE that the numerical method under consideration approximates more accurately. For the theory, we need a second-order modified equation for the doubled-up system and this contains second-order information about the coupling of the fine and coarse levels. In particular, we consider modified equations for the double-up system (3.1) of the form:

$$\begin{aligned} d\boldsymbol{X}_h &= \big[\boldsymbol{f}(\boldsymbol{X}_h) + h\boldsymbol{f}_1(\boldsymbol{X}_h)\big]\,dt + \big[G(\boldsymbol{X}) + hG_1(\boldsymbol{X}_h)\big]\,d\boldsymbol{W}(t), & \boldsymbol{X}(0) &= \boldsymbol{X}_0, \\ d\boldsymbol{Y}_h &= \big[\boldsymbol{f}(\boldsymbol{Y}_h) + h\boldsymbol{f}_2(\boldsymbol{Y}_h)\big]\,dt + \big[G(\boldsymbol{Y}_h) + hG_2(\boldsymbol{Y}_h)\big]\,d\boldsymbol{W}(t), & \boldsymbol{Y}(0) &= \boldsymbol{X}_0, \end{aligned} \tag{3.3}$$

for $\boldsymbol{f}_i : \mathbb{R}^d \to \mathbb{R}^d$ and $\boldsymbol{G}_i : \mathbb{R}^d \to \mathbb{R}^{d \times m}$ for $i = 1, 2$. (This could be extended to allow $\boldsymbol{f}_i, G_i$ to depend on both $\boldsymbol{X}_h$ and $\boldsymbol{Y}_h$.) When the same integrator is used for each component, but with time steps h and $h/2$, it must hold that $\boldsymbol{f}_2 = \boldsymbol{f}_1/2$ and $G_2 = G_1/2$. We show in Theorem 3.4 that, subject to regularity conditions on the coefficients, the MLMC complexity theorem applies if a second-order modified equation exists and therefore MLMC works with $\mathcal{O}(\epsilon^{-2})$ complexity.

The additional difficulty is that $\boldsymbol{X}_h \neq \boldsymbol{Y}_h$ and we must estimate the variance of $\phi(\boldsymbol{X}_h) - \phi(\boldsymbol{Y}_h)$. We use a mean-square analysis and the following lemma, which gives a first-order $L^2(\Omega, \mathbb{R}^d)$ bound on $\boldsymbol{Z}(t) - \boldsymbol{Z}_h(t)$. The lemma requires a number of regularity assumptions on the coefficients of the modified equation, which hold, for example, if $\boldsymbol{f}, \boldsymbol{f}_i$ and G, G_i are globally Lipschitz continuous.

Lemma 3.3. *For $t \in [0, T]$, let $\boldsymbol{Z}(t)$ satisfy the Ito SDE (3.1) and $\boldsymbol{Z}_h(t) = [\boldsymbol{X}_h(t), \boldsymbol{Y}_h(t)]$ satisfy the modified equation (3.3). Suppose that*

(i) *$\boldsymbol{f} \colon \mathbb{R}^d \to \mathbb{R}^d$ and $G \colon \mathbb{R}^d \to \mathbb{R}^{d \times m}$ are globally Lipschitz continuous with Lipschitz constant $L > 0$.*

(ii) *There exists $C_1 > 0$ such that, for all $h > 0$ sufficiently small,*

$$\mathbb{E}\Big[\big\|\boldsymbol{f}_1(\boldsymbol{X}_h(s))\big\|^2\Big], \mathbb{E}\Big[\big\|G_1(\boldsymbol{X}_h(s))\big\|_{\mathrm{F}}^2\Big] \leq C_1,$$
$$\mathbb{E}\Big[\big\|\boldsymbol{f}_2(\boldsymbol{Y}_h(s))\big\|^2\Big], \mathbb{E}\Big[\big\|G_2(\boldsymbol{Y}_h(s))\big\|_{\mathrm{F}}^2\Big] \leq C_1, \qquad s \in [0, T],$$

where $\|\cdot\|_{\mathrm{F}}$ denotes the Frobenius norm.

Then, if $\psi \colon \mathbb{R}^{2d} \to \mathbb{R}$ is globally Lipschitz continuous, we have, for some constant $C_2 > 0$ independent of h,

$$\mathbb{E}\Big[\big|\psi(\boldsymbol{Z}(t)) - \psi(\boldsymbol{Z}_h(t))\big|^2\Big] \leq C_2 h^2, \qquad \text{for } t \in [0, T].$$

Proof. This is an elementary calculation with the Gronwall inequality and Ito isometry. See Appendix A. $\qquad\square$

We are now able to state and prove the main theorem of this article. In contrast to Theorem 3.2, ϕ is assumed to be Lipschitz here.

Theorem 3.4. *Fix $T > 0$. Let $\phi \in C_{\mathrm{poly}}^\infty(\mathbb{R}^d)$ be globally Lipschitz continuous. Suppose that*

(i) *$\boldsymbol{X}_n$ and $\boldsymbol{Y}_n$ are weak order-α approximations to $\boldsymbol{X}(t)$ for some $\alpha > 1/2$,*

(ii) *$\boldsymbol{Z}_n$ are second-order weak approximations to $\boldsymbol{Z}_h(t)$, and*

(iii) *the assumptions of Lemma 3.3 hold.*

Then Conditions (i)–(iii) of Theorem 2.1 hold with $[\mathcal{P}_\ell^{(i)}, \mathcal{P}_{\ell-1}^{(i)}]$ given by iid *samples of $[\phi(\boldsymbol{X}_M), \phi(\boldsymbol{Y}_M)]$ with $h = h_\ell$.*

Proof. As before, conditions (i) and (ii) are straightforward. It is the third condition, which normally follows from a strong-approximation theory, that requires the modified equation. Let $\psi(\boldsymbol{Z}) \coloneqq \phi(\boldsymbol{X}) - \phi(\boldsymbol{Y})$ and note that $\psi^2 \in C_{\mathrm{poly}}^\infty(\mathbb{R}^{2d})$. As $\boldsymbol{Z}_n$ is a second-order weak approximation to $\boldsymbol{Z}_h(t)$, we have

$$\mathbb{E}\big[\psi(\boldsymbol{Z}_M)^2 - \psi(\boldsymbol{Z}_h(T))^2\big] = \mathcal{O}\big(h^2\big).$$

By definition of ψ,

$$\mathbb{E}\Big[\big|\phi(\boldsymbol{X}_M) - \phi(\boldsymbol{Y}_M)\big|^2 - \big|\phi(\boldsymbol{X}_h(T)) - \phi(\boldsymbol{Y}_h(T))\big|^2\Big] = \mathcal{O}\big(h^2\big). \tag{3.4}$$

Using the fact that $\boldsymbol{X}(t) = \boldsymbol{Y}(t)$ a.s.,

$$\mathbb{E}\Big[\big|\phi(\boldsymbol{X}_h(T)) - \phi(\boldsymbol{Y}_h(T))\big|^2\Big] = \mathbb{E}\Big[\big|\phi(\boldsymbol{X}_h(T)) - \phi(\boldsymbol{X}(T)) + \phi(\boldsymbol{Y}(T)) - \phi(\boldsymbol{Y}_h(T))\big|^2\Big]$$
$$= \mathbb{E}\Big[\big|\psi(\boldsymbol{Z}_h(T)) - \psi(\boldsymbol{Z}(T))\big|^2\Big].$$

8

Lemma 3.3 applies and the right-hand side in the last equation is $\mathcal{O}(h^2)$. Consequently,

$$\mathbb{E}\left[\left|\phi(\boldsymbol{X}_h(T)) - \phi(\boldsymbol{Y}_h(T))\right|^2\right] = \mathcal{O}(h^2). \tag{3.5}$$

Together, Eqs. (3.4) and (3.5) imply that

$$\mathbb{E}\left[\left|\phi(\boldsymbol{X}_M) - \phi(\boldsymbol{Y}_M)\right|^2\right] = \mathcal{O}(h^2). \tag{3.6}$$

The remainder of the proof is the same as for Theorem 3.2. $\qquad\square$

By taking $\boldsymbol{X}$ to be the exact solution (i.e., $\boldsymbol{X}_n = \boldsymbol{X}(t_n)$) and $\phi\colon \mathbb{R}^d \to \mathbb{R}$ as a projection onto the ith coordinate, Eq. (3.6) implies that $\|\boldsymbol{X}(T) - \boldsymbol{Y}_M\|_{L^2(\Omega, \mathbb{R}^d)} = \mathcal{O}(h)$ and hence first-order strong convergence can be proved by this method. This is consistent with the observation that the Euler–Maruyama method, which is not first-order strongly convergent in general, does not have a second-order modified equation.

In summary, subject to smoothness conditions, if MLMC is applied with an integrator that has a second-order modified equation like Eq. (3.3) then the variance of the coarse–fine correction is $\mathcal{O}(h^2)$ and the complexity of MLMC is $\mathcal{O}(\epsilon^{-2})$. Though the rate is fixed, the complexity of MLMC depends on the specific integrator used through the constant and, as we now show, this leads to large variations in efficiency.

4 Application to the Langevin equation

Before showing how they can be used for MLMC, we introduce several integrators for the Langevin equation.

4.1 Splitting methods

Splitting methods are an important class of numerical integrators for differential equations. In the case of ODEs, they allow the vector field to be broken down into meaningful parts and integrated separately over a single time step, before combining into an integrator for the full vector field. See for example (Hairer *et al.*, 2010; Leimkuhler & Reich, 2004). The Langevin equation breaks down into the sum of a Hamiltonian system and a linear SDE for an Ornstein–Uhlenbeck (OU) process. Then, for a splitting method, we define symplectic integrators for the Hamiltonian system

$$\begin{aligned} \frac{d\boldsymbol{Q}}{dt} &= \boldsymbol{P}, \\ \frac{d\boldsymbol{P}}{dt} &= -\nabla V(\boldsymbol{Q}). \end{aligned} \tag{4.1}$$

The OU process $\boldsymbol{P}(t)$, which satisfies

$$d\boldsymbol{P} = -\lambda \boldsymbol{P}\, dt + \sigma\, d\boldsymbol{W}(t), \tag{4.2}$$

can be integrated exactly and we use this fact to define a so-called geometric integrator for Eq. (4.2). It is clear that the sum of the right-hand sides of these two systems gives Eq. (1.5). There are a number of ways of combining integrators of Eqs. (4.1) and (4.2) to define an integrator of the full system. The simplest, also known as the Lie–Trotter splitting, is to simulate Eqs. (4.1) and (4.2) alternately on time intervals of length h. In general, this technique can only be first-order accurate in the weak sense. Alternatively, if the underlying integrators are second order, we can define a second-order splitting method by applying Eq. (4.2) on a half step, then Eq. (4.1) for a full step, and finally apply again Eq. (4.2) on a half step. This is called the symmetric Strang splitting. See also (Leimkuhler *et al.*, 2013).

We now define specific integrators for Eqs. (4.1) and (4.2). Eq. (4.1) is a separable Hamiltonian system, and the symplectic Euler method and Störmer–Verlet methods provide

simple, explicit methods for its numerical solution. The symplectic Euler method is first-order accurate and the Störmer–Verlet method is second-order accurate.

The solution of Eq. (4.2) is a multi-dimensional OU process and can be written as

$$\boldsymbol{P}(t) = \mathrm{e}^{-\lambda t} \boldsymbol{P}(0) + \sigma \boldsymbol{I}(0, t), \qquad \boldsymbol{I}(t_1, t_2) \coloneqq \int_{t_1}^{t_2} \mathrm{e}^{-\lambda(t_2 - s)} \, d\boldsymbol{W}(s). \tag{4.3}$$

Each component of $\boldsymbol{I}$ is *iid* with mean zero and variance

$$\operatorname{Var} I_i(t_1, t_2) = \int_{t_1}^{t_2} \mathrm{e}^{-2\lambda(t_2 - s)} \, ds = \frac{1 - e^{-2\lambda(t_2 - t_1)}}{2\lambda}, \tag{4.4}$$

so that $\boldsymbol{I}(t_1, t_2) \sim \mathrm{N}(\boldsymbol{0}, \alpha_{t_2 - t_1}^2 I)$ for $\alpha_t \coloneqq \sqrt{(1 - \mathrm{e}^{-2\lambda t})/2\lambda}$. This suggests taking the following as the numerical integrator: for a time step $h > 0$,

$$\boldsymbol{P}_{n+1} = \mathrm{e}^{-\lambda h} \boldsymbol{P}_n + \sigma \alpha_h \boldsymbol{\xi}_n \tag{4.5}$$

for $\boldsymbol{\xi}_n \sim \mathrm{N}(0, I)$ *iid*. If $\boldsymbol{P}_n = \boldsymbol{P}(t_n)$, then $\boldsymbol{P}_{n+1}$ has the same distribution as $\boldsymbol{P}(t_{n+1})$ and this method is exact in the sense of distributions. Methods of this type, where the variation of constants formula (4.3) is used for the discretisation, are often called geometric integrators (Bou-Rabee & Owhadi, 2009).

The full equations for the first order splitting (symplectic Euler) and second-order splitting (Störmer–Verlet) are written as follows:

Symplectic Euler/OU   For $\boldsymbol{\xi}_n$ *iid* with distribution $\mathrm{N}(\boldsymbol{0}, I)$,

$$\begin{aligned}
\boldsymbol{P}_{n+1}^* &= \mathrm{e}^{-\lambda h} \boldsymbol{P}_n + \sigma \alpha_h \boldsymbol{\xi}_n, \\
\boldsymbol{P}_{n+1} &= \boldsymbol{P}_{n+1}^* - h \, \nabla V(\boldsymbol{Q}_n), \\
\boldsymbol{Q}_{n+1} &= \boldsymbol{Q}_n + \boldsymbol{P}_{n+1} \, h.
\end{aligned} \tag{4.6}$$

Störmer–Verlet/OU   For $\boldsymbol{\xi}_n, \boldsymbol{\xi}_{n+1/2}$ *iid* with distribution $\mathrm{N}(\boldsymbol{0}, I)$

$$\begin{aligned}
\boldsymbol{P}_{n+1/2}^* &= \mathrm{e}^{-\lambda h/2} \boldsymbol{P}_n + \sigma \alpha_{h/2} \boldsymbol{\xi}_n, \\
\boldsymbol{P}_{n+1/2} &= \boldsymbol{P}_{n+1/2}^* - \frac{1}{2} h \, \nabla V(\boldsymbol{Q}_n), \\
\boldsymbol{Q}_{n+1} &= \boldsymbol{Q}_n + h \boldsymbol{P}_{n+1/2}, \\
\boldsymbol{P}_{n+1}^* &= \boldsymbol{P}_{n+1/2} - \frac{1}{2} h \, \nabla V(\boldsymbol{Q}_{n+1}), \\
\boldsymbol{P}_{n+1} &= \mathrm{e}^{-\lambda h/2} \boldsymbol{P}_{n+1}^* + \sigma \alpha_{h/2} \boldsymbol{\xi}_{n+1/2}.
\end{aligned} \tag{4.7}$$

Subject to regularity conditions on the coefficients, Eq. (4.6) is first-order and Eq. (4.7) second-order accurate in the weak sense by application of the Baker–Campbell–Hausdorff formula.

4.2   Modified equations for the Langevin equation

Consider the Langevin equation (1.5). Following (Shardlow, 2006; Zygalakis, 2011) by using a computer algebra system to verify consistency of moments to fifth order, it is easy to find modified equations for the numerical integrators developed in §4.1. For example, for the first-order splitting method with $d = 1$, the doubled-up modified equation is as follows: Denote by $[Q_n, P_n]$ the numerical approximation on the coarse level (step h) and $[q_n, p_n]$ on the fine

level (step $h/2$). The second-order modified equation is

$$dQ = \left[P - \frac{1}{2}h\big(V'(Q) + \lambda P\big) \right] dt + \sigma \frac{1}{2}h\, dW(t),$$

$$dP = \left[-\lambda P - V'(Q) - \frac{1}{2}h\big(\lambda V'(Q) - PV''(Q)\big) \right] dt + \sigma\, dW(t),$$

$$dq = \left[p - \frac{1}{4}h\big(V'(q) + \lambda p\big) \right] dt + \sigma \frac{1}{4}h\, dW(t),$$

$$dp = \left[-\lambda p - V'(q) - \frac{1}{4}h\big(\lambda V'(q) - pV''(q)\big) \right] dt + \sigma\, dW(t),$$

(4.8)

where $W(t)$ is the same Brownian motion for p and P. We conclude then that this method leads to $\mathcal{O}(h^2)$ variances in the coarse–fine correction, if the coefficients are sufficiently well behaved. Identifying when the coefficients are well behaved is hard. For example, it is sufficient that the drift and diffusion in both the original and modified equations are globally Lipschitz. These however are very strong conditions and do not hold for many realistic potentials.

For the second-order splitting method (based on Störmer–Verlet method and exact OU integration), we can apply Theorem 3.2 to see that the variance of the coarse–fine corrections is $\mathcal{O}(h^2)$. The regularity condition is on the original drift and diffusion and holds if $\nabla V \colon \mathbb{R}^d \to \mathbb{R}^d$ is sufficiently smooth (e.g., infinitely differentiable and Lipschitz).

4.3 MLMC with splitting methods

Let $\boldsymbol{X} = [\boldsymbol{Q}, \boldsymbol{P}]$ denote the state-space variable. A key step in MLMC is computing approximations to $\boldsymbol{X}(t_{n+2})$ at $t_n = nh$ given $\boldsymbol{X}(t_n)$ based on integrators with time steps h and $2h$ that are coupled so the difference between the approximations has small variance. For the Euler–Maruyama method, this is achieved by choosing increments $\Delta \boldsymbol{W}_n, \Delta \boldsymbol{W}_{n+1}$ for the computation with time step $h/2$, and choosing the sum $\Delta \boldsymbol{W}_n + \Delta \boldsymbol{W}_{n+1}$ for the corresponding interval of the computation with time step h.

It is hard to sample $\boldsymbol{I}(0, t)$ in Eq. (4.5) based on increments of the particular sample path of $\boldsymbol{W}(t)$ and, as a method for strong approximation, it is limited. It is easy however to sample $\boldsymbol{I}(0, t)$ as a Gaussian random variable. We now show how to couple fine–coarse integrators for the MLMC method, without the direct link to the increment. First, note that

$$\boldsymbol{I}(0, 2h) = \int_0^{2h} \mathrm{e}^{-\lambda(2h-s)}\, d\boldsymbol{W}(s)$$

$$= \int_0^{h} \mathrm{e}^{-\lambda(2h-s)}\, d\boldsymbol{W}(s) + \int_h^{2h} \mathrm{e}^{-\lambda(2h-s)}\, d\boldsymbol{W}(s)$$

$$= rI(0, h) + I(h, 2h), \qquad r := \mathrm{e}^{-\lambda h}.$$

$$\boldsymbol{I}(0, h), \boldsymbol{I}(h, 2h) \sim \mathrm{N}\big(0, \alpha_h^2 I\big) \ iid.$$

We can simulate $\boldsymbol{I}(0, h)$ and $\boldsymbol{I}(0, 2h)$, by generating $\boldsymbol{\xi}_i \sim \mathrm{N}(\boldsymbol{0}, I)$ *iid* and computing

$$\boldsymbol{I}(0, h) = \alpha_h \boldsymbol{\xi}_1, \qquad \boldsymbol{I}(h, 2h) = \alpha_h \boldsymbol{\xi}_2.$$

As $\alpha_h^2 = (1 - r^2)/2\lambda$ and $\alpha_{2h}^2 = (1 - r^4)/2\lambda$, we have $\alpha_h^2(1 + r^2) = \alpha_{2h}^2$. Then,

$$\boldsymbol{I}(0, 2h) = \alpha_{2h} \frac{1}{\sqrt{1 + r^2}}(r\boldsymbol{\xi}_1 + \boldsymbol{\xi}_2).$$

(4.9)

Given $\boldsymbol{P}_n$ at time t_n, we find $\boldsymbol{P}_{n+2}$ using two time steps of size h by

$$\boldsymbol{P}_{n+1} = \mathrm{e}^{-\lambda h}\boldsymbol{P}_n + \sigma \alpha_h \boldsymbol{\xi}_n$$

$$\boldsymbol{P}_{n+2} = \mathrm{e}^{-\lambda h}\boldsymbol{P}_{n+1} + \sigma \alpha_h \boldsymbol{\xi}_{n+1},$$

11

for $\boldsymbol{\xi}_n \sim \mathrm{N}(0, I)$ *iid*. This is equivalent to a single time step of size $2h$ and

$$\boldsymbol{P}_{n+2} = \mathrm{e}^{-2\lambda h} \boldsymbol{P}_n + \sigma \alpha_{2h} \boldsymbol{\xi}_n^*, \qquad \boldsymbol{\xi}_n^* := \frac{r \boldsymbol{\xi}_n + \boldsymbol{\xi}_{n+1}}{\sqrt{r^2 + 1}}.$$

This method is used to generate the increments when using splitting methods within MLMC.

5 Numerical experiments

We developed an object-oriented C++ code to compare the performance of different numerical methods for two model problems. The modular structure of the templated code makes it easy to change key components, such as the time-stepping method or random-number distribution, without negative impacts on the performance. The source code is available under the LGPL 3 license as a git repository on `https://bitbucket.org/em459/mlmclangevin`[1].

Key to the choice of parameters in Algorithm 1 is the balance between bias error and statistical error. We assume that the bias error has the form in Theorem 2.1(i) for a proportionality constant c_1 and that the finest time step $h_L = T/(M_0 2^L)$. Then, for a bias error of size $\epsilon/\sqrt{2}$, we require that

$$c_1 \left(\frac{T}{M_0 2^L} \right)^\alpha = \frac{\epsilon}{\sqrt{2}}.$$

Given c_1, α, and T as well as a choice for M_0, this can be solved to determine ϵ from L or vice versa. The constant c_1 can be approximated by assuming that $\mathbb{E}\left[\widehat{\mathcal{P}}_\ell - \mathcal{P}\right] = \tilde{c}_1 h_\ell^\alpha$ for some $\tilde{c}_1 \in \mathbb{R}$, so that

$$\widehat{Y}_{\ell, N_\ell} \approx \tilde{c}_1 h_\ell^\alpha - \tilde{c}_1 h_{\ell-1}^\alpha = \tilde{c}_1 (1 - 2^\alpha) h_\ell^\alpha$$

and calculating $c_1 = |\tilde{c}_1|$ after computing the left-hand side numerically.

The following integrators are used in the numerical experiments below:

EMG and EMG+ Euler–Maruyama as given by Eq. (1.2) with $M_0 = 4$ (EMG) and $M_0 = 8$ (EMG+).

SEG First-order splitting method with symplectic Euler/exact OU and $M_0 = 4$. See Eq. (4.6).

SVG Second-order splitting method with Störmer–Verlet/exact OU and $M_0 = 4$. See Eq. (4.7).

Richardson extrapolation is a well-known technique for increasing the accuracy of a numerical approximation by computing two approximations with different discretisation parameters and taking a linear combination that eliminates the lowest-order term for the error. Its extension to SDEs was developed by (Talay & Tubaro, 1990) and is particularly convenient for use with MLMC, as MLMC computes approximations on several levels and this has already been explored in (Giles, 2008). Thus, we take $\widehat{\mathcal{P}}_L$ and $\widehat{\mathcal{P}}_{L-1}$ and suppose that, for some constants $\tilde{c}_1$ and $\alpha' > \alpha$,

$$\mathbb{E}\left[\widehat{\mathcal{P}}_L\right] = \mathbb{E}[\mathcal{P}] + \tilde{c}_1 h_L^\alpha + \mathcal{O}\left(h_L^{\alpha'}\right)$$
$$\mathbb{E}\left[\widehat{\mathcal{P}}_{L-1}\right] = \mathbb{E}[\mathcal{P}] + \tilde{c}_1 h_{L-1}^\alpha + \mathcal{O}\left(h_L^{\alpha'}\right).$$

A simple linear combination of the two gives a higher-order approximation to $\mathbb{E}[\mathcal{P}]$; in particular, for SEG, we have $\alpha = 1$ and $\alpha' = 2$ and

$$2\mathbb{E}\left[\widehat{\mathcal{P}}_L\right] - \mathbb{E}\left[\widehat{\mathcal{P}}_{L-1}\right] = \mathbb{E}[\mathcal{P}] + \mathcal{O}\left(h_\ell^2\right).$$

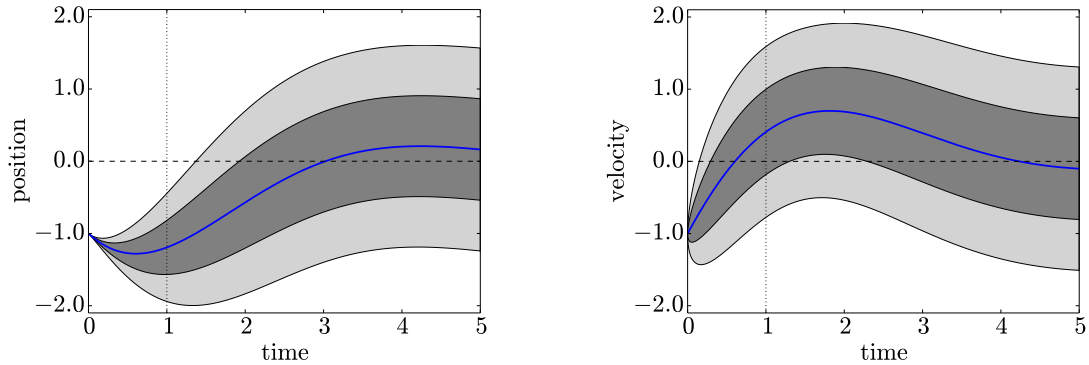[1] All enquiries about the code should be addressed to `e.mueller@bath.ac.uk`.

Figure 1: The position (left) and velocity (right) of a randomly forced harmonic oscillator with damping for $\omega_0 = 1 = \lambda = \sigma$. The mean value is shown together with one (dark gray) and two standard deviations (light gray).

An approximation to the left-hand side is given by $\widehat{\mathcal{P}}^{(\mathrm{MLMC})} + \widehat{Y}_{L,N_L}$. For SVG, we have $\alpha = 2$ and $\alpha' = 4$, and

$$\frac{1}{3}\left(4\mathbb{E}\left[\widehat{\mathcal{P}}_L\right] - \mathbb{E}\left[\widehat{\mathcal{P}}_{L-1}\right]\right) = \mathbb{E}[\mathcal{P}] + \mathcal{O}\left(h_\ell^4\right).$$

An approximation to the left-hand side is given by $\widehat{\mathcal{P}}^{(\mathrm{MLMC})} + \frac{1}{3}\widehat{Y}_{L,N_L}$. To observe the improved accuracy, the statistical error must also be reduced to match the bias error. An increase in accuracy from second- to fourth-order accuracy is achieved because the integrator is symmetric.

In the experiments, we apply extrapolation in the following scenarios:

EMGe and EMGe+ EMG/EMG+ with extrapolation, increasing the weak order of convergence from one to two.

SEGe SEG with extrapolation, again increasing the weak order of convergence from one to two.

SVGe SVG with extrapolation, increasing the weak order of convergence from two to four. Due to the fourth-order convergence, it is sufficient to take large time steps, and we choose $L = 2$ and vary M_0 rather than L.

5.1 Langevin equation for the damped harmonic oscillator

We first consider Eq. (1.5) with $d = 1$ and

$$V(Q) = \frac{1}{2}\omega_0^2 Q^2. \tag{5.1}$$

Physically, with this potential, Eq. (1.5) describes a randomly forced harmonic oscillator with resonance frequency ω_0 and damping parameter λ; the strength of the Gaussian forcing is given by σ. For $\omega_0 = 0$ (i.e., in the absence of a potential), the SDE can be interpreted as a model for the dispersion of an atmospheric pollutant in a one-dimensional turbulent velocity field (see Rodean (1996)). In this case, $\sigma^2/(2\lambda)$ is the turbulent-velocity variance and $1/\lambda$ the velocity relaxation-time. In Figure 1, the marginal distributions for the position and velocity are visualised as a function of t for the first set of parameters used in the numerical experiments ($\omega_0 = 1 = \lambda = \sigma$ and $P(0) = Q(0) = -1$).

We choose this simple example, for which we know the analytical solution, to verify the correctness of our code and to quantify numerical errors; exact solutions of the Langevin equation are also described in (Risken, 1996). As the system is linear, the joint pdf of Q and P

13

is Gaussian and is defined by their mean and covariance. Denoting $\boldsymbol{X}(t) = (Q(t), P(t))^\mathsf{T}$ and the initial solution by $\boldsymbol{X}_0 = \boldsymbol{X}(t=0) = (Q(t=0), P(t=0))^\mathsf{T}$, we have

$$\boldsymbol{X}(t) = \exp\big[-\Lambda t\big]\boldsymbol{X}_0 + \int_0^t \exp\big[-\Lambda(t-s)\big]\boldsymbol{\Sigma}\, dW(s) \tag{5.2}$$

with

$$\Lambda := \begin{pmatrix} 0 & -1 \\ \omega_0^2 & \lambda \end{pmatrix}, \qquad \boldsymbol{\Sigma} := \begin{pmatrix} 0 \\ \sigma \end{pmatrix}.$$

$\boldsymbol{X}(t)$ follows a Gaussian distribution with mean

$$\mathbb{E}\big[\boldsymbol{X}(t)\big] = \exp\big[-\Lambda t\big]\boldsymbol{X}_0 \tag{5.3}$$

and covariance matrix

$$B(t) := \int_0^t \exp\big[-\Lambda(t-s)\big]\boldsymbol{\Sigma}\boldsymbol{\Sigma}^\mathsf{T}\exp\big[-\Lambda^\mathsf{T}(t-s)\big]\, ds\,, \tag{5.4}$$

which can easily be evaluated using a computer algebra system.

Numerical results

We compute $\mathbb{E}[\phi(\boldsymbol{X}(1))]$ for $\phi(Q, P) = \exp(-2(P-0.5)^2)\sqrt{2/\pi}$ and the following set of parameters:

1. $\omega_0 = 1$, $\lambda = 4$ and $\sigma = 2$.

2. $\omega_0 = 1$, $\lambda = 9$ and $\sigma = 3$.

The initial position and velocity were set to $Q(t=0) = P(t=0) = -1$ in both cases. Errors are computed using the exact value computed from Eqs. (5.3) and (5.4). The exact values are 0.447904416997582 and 0.418086875513087, respectively. The CPU time scaled by ϵ^{-2} and the error (bias error plus one standard deviation) scaled by ϵ are plotted in Figures 2 and 3 against ϵ. The scaling means we expect both graphs to be flat. We observe for both parameter sets that the integrators based on the exact OU process are the most efficient for small ϵ. Even though SVGe uses a weak fourth-order accurate integrator, the complexity of MLMC cannot be reduced beyond $\mathcal{O}(\epsilon^{-2})$ and it is the same as for the other integrators. The improvements come by improving constants, in this case by about a factor 4 in comparison to EMG. For the second set of parameter values in Figure 3, the relaxation time is shorter and the noise is larger, and the improvement due to the splitting methods is even more pronounced (factor 10).

In order to take large time-steps, it is necessary to ensure the stability of the integrator. It is well known from deterministic differential equations that most explicit integrators will have a stability constraint on the time-step size. This is the same for SDEs and such stability constraints may severely restrict the number of levels that can be employed in the MLMC method and thus its efficiency (Abdulle & Blumenthal, 2013; Hutzenthaler *et al.*, 2013). Exact sampling of the Ornstein–Uhlenbeck process poses no stability constraints, allowing for smaller values of M_0 and thus for larger numbers of levels in MLMC in the case of splitting methods. For example, in the above simulations, increasing the number of time steps from $M_0 = 4$ to $M_0 = 8$ in Euler-Maruyama (cf. EMG and EMG+, as well as EMGe and EMGe+) lead to an improvement in efficiency. The same change has no effect in SEG. However, the symplectic methods we are using for the Hamiltonian part are explicit and have their own stability constraint (Skeel & Izaguirre, 2002), somewhat limiting this benefit of splitting methods.

5.2 Double-well potential

We now change the potential and consider the double-well potential

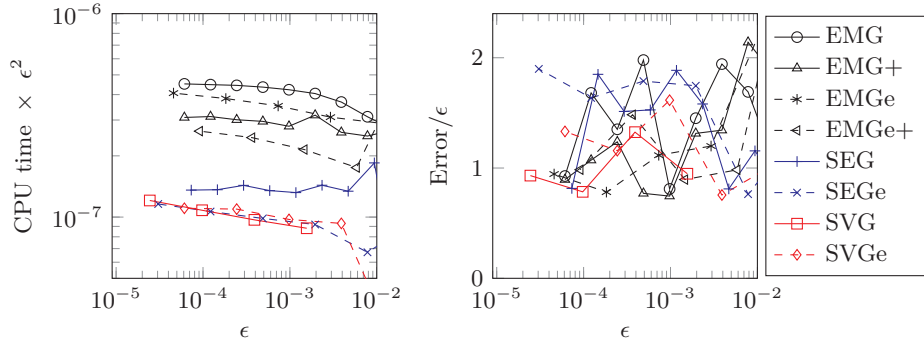$$V(Q) = \frac{\omega_0^2}{8Q_{\min}^2}(Q^2 - Q_{\min}^2)^2,$$

Figure 2: For the harmonic oscillator with parameter set 1. The left-hand plot shows the CPU time for a given value of ϵ; the time is scaled by ϵ^{-2} and this leads to a nearly flat profile in each case. The right-hand plot shows the bias error plus one standard deviation; the errors are divided by ϵ to show both mean and standard deviation are $\mathcal{O}(\epsilon)$.
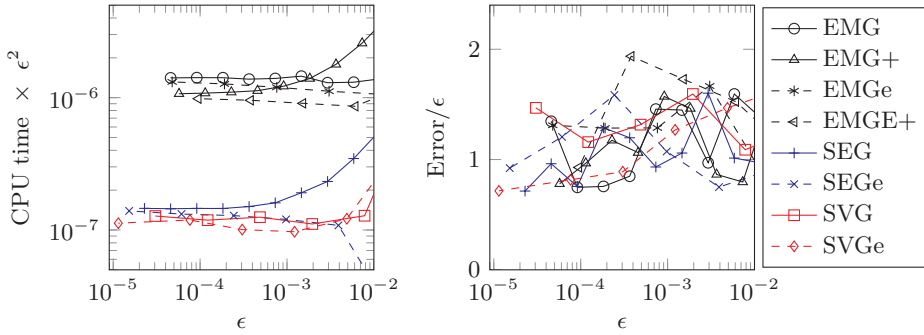

Figure 3: For the harmonic oscillator with parameter set 2. Compared to Figure 2, the difference between the splitting methods and Euler–Maruyama is significantly larger.

where $Q_{\min}$ and ω_0 are parameters. We compute $\mathbb{E}[\phi(\boldsymbol{X}(T))]$ for $\phi(Q, P) \coloneqq (Q + Q_{\min})^2 + P^2$ (note $(Q + Q_{\min})^2$ takes distinct values at the bottom of the wells $Q = \pm Q_{\min}$).

For the numerical experiments in Figure 4, we choose parameter values $Q_{\min} = \omega_0 = 1$, $\lambda = 2$, $\sigma = 4$, and take initial data $Q(t = 0) = P(t = 0) = -1$. The scaled CPU time and error for $T = 1$ are plotted against ϵ in Figure 4, where errors are computed relative to a numerically computed value given by 4.52782626985. It is noticeable again that the splitting methods and especially the symplectic Euler-based methods are most efficient.

In Figure 5, we explore the behaviour of the algorithm as we increase the length of the time interval T. For the plot, we scale the CPU time by $\epsilon^{-2}T$; the computation time scales linearly with the number of time steps and, by scaling by T, we see how the MLMC algorithm behaves with increasing T. The errors are computed relative to the numerically computed values 6.11075602345 for $T = 2$; 7.11570774835 for $T = 4$; and 7.2125872733 for $T = 8$. The values for $T = 4$ and $T = 8$ are close, which indicates the system has moved close to the invariant measure by this time. In each case, SEG is most efficient and we see the measure of CPU time $\times \epsilon^2/T$ decrease from about 5×10^{-5} for $T = 1$ to about 10^{-4} for $T = 8$. The profiles are also less flat as T is increased, indicating that the time steps may not be small enough to have entered the asymptotic regime. It is natural that the gains become less pronounced, when we come close to the invariant measure and the coupling between levels has decayed.

6 Further enhancements

Theorems 3.2 and 3.4 provide a route to analysing the MLMC entirely by weak-approximation properties of the numerical method. This has a number of advantages: from the theoretical
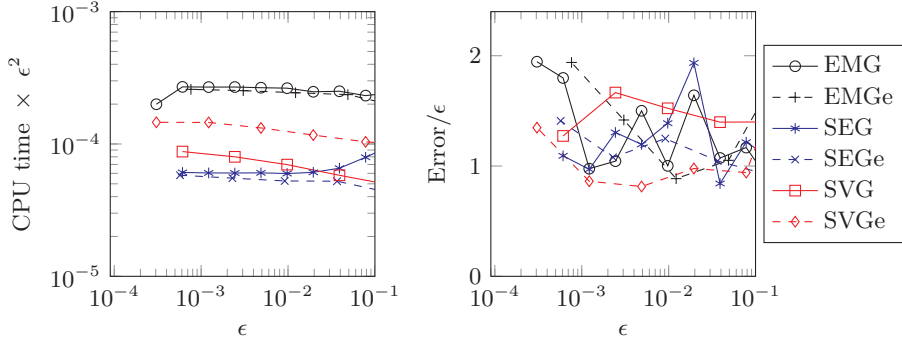
15

Figure 4: Numerical results for the double-well potential (plots as above).
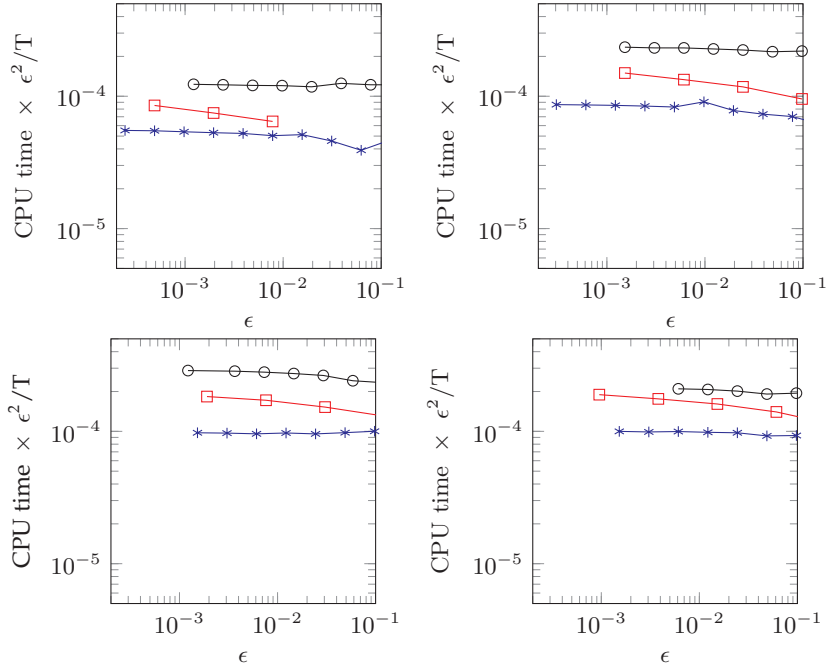

Figure 5: From top-left to bottom-right, plots of CPU time scaled by $T\epsilon^{-2}$ for $T = 1, 2, 4, 8$ and the double-well potential with $\omega_0 = 1$, $\lambda = 2$, $\sigma = 4$.

point of view, the analysis works for a wider set of test functions compared to the analysis of (Giles, 2008), which demands that quantities of interest ϕ are globally Lipschitz continuous. From the algorithmic point of view, the order of weak convergence is determined by moment conditions up to a given degree depending on the order of convergence. There are a number of ways to satisfy these conditions. It is widely known (Kloeden & Platen, 1992) that the Gaussian random variables can be replaced by discrete random variables without disturbing the weak order of convergence. The obvious question then is whether we can use discrete random variables to our advantage also in the context of MLMC.

MLMC depends crucially on the fact that the sum of two independent Gaussian random variables is also Gaussian. This allows increments to be generated on the fine levels and combined to give a random variable with the same distribution on the next coarser level, using Eq. (4.9). Discrete variables do not have this property. While Theorem 3.4 implies the coupling condition of Theorem 2.1(iii), the sum of two three-point random variables is not a three-point random variable and the telescoping sum breaks down. In general, using discrete random variables with MLMC introduces extra error due to the telescoping sum no longer being exact. Though (Belomestny & Nagapetyan, 2014) provides an approach that preserves the telescoping

sum by using a different discrete random variable on each level. Here we do not follow this route. Instead, we use the same discrete random variable on each level, accepting the additional bias error that this introduces, which crucially is of higher order. To control this additional bias and to ensure the total error is still below our chosen tolerance, we change the number of levels L and the coarsest mesh size h_0. Discrete random variables are cheaper to generate than Gaussian random variables and the coarsest level can be evaluated exactly, which we exploit to achieve a significant speed-up in the small noise case.

6.1 Random variables with discrete distribution

The modified equations are unchanged if the Gaussian random variables in the integrator are replaced by random variables with the same moments to order five (including all cross moments to order five arising from the doubled-up system). For example, we can replace samples of *iid* $N(0,1)$ random variables by *iid* samples of the random variable ζ with distribution

$$\mathbb{P}\big(\zeta = 0\big) = \frac{2}{3}, \qquad \mathbb{P}\big(\zeta = \pm\sqrt{3}\big) = \frac{1}{6}; \tag{6.1}$$

or

$$\mathbb{P}\Big(\zeta = \pm\sqrt{3+\sqrt{6}}\Big) = c, \qquad \mathbb{P}\Big(\zeta = \pm\sqrt{3-\sqrt{6}}\Big) = \frac{1}{2} - c, \qquad c := \frac{1}{2}\Big(1 - \frac{3+\sqrt{6}}{6}\Big). \tag{6.2}$$

We refer to ζ as the three- and four-point approximations to the Gaussian, respectively. This is a well-known trick for weak approximation of SDEs, e.g. (Kloeden & Platen, 1992, §14.2). The approximations have a number of advantages, as ζ is quicker to sample than a Gaussian and, due to the finite number of states, averages of functionals of ζ can be computed exactly.

6.2 Exact evaluation of the coarse-level expectation

For all our integrators, the evaluation of the coarse-level estimator $\widehat{\mathcal{P}}_0$ with time step $h_0 = T/M_0$ is the computationally most expensive part of the MLMC algorithm: even though the number of time steps and hence the number of samples per path is small, a large number of individual paths needs to be evaluated to reduce the variance of the coarse-level estimator. This cost can be reduced dramatically if a discrete distribution as discussed in §6.1 is used for the individual samples $\boldsymbol{\xi}_n$. In this case, a significantly cheaper estimator, which does not rely on Monte Carlo sampling, can be constructed. If the random numbers $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{M_0}$ for each path are drawn from the three-point approximation in Eq. (6.1), there is only a finite number $n_{\boldsymbol{\xi}}$ of possible samples $\boldsymbol{\xi}^{(i)} = \{\boldsymbol{\xi}_1^{(i)}, \ldots, \boldsymbol{\xi}_{M_0}^{(i)}\}$, each with associated probability $\mathbb{P}(\boldsymbol{\xi}^{(i)}) = \mathbb{P}(\boldsymbol{\xi}_1 = \boldsymbol{\xi}_1^{(i)}) \cdots \mathbb{P}(\boldsymbol{\xi}_{M_0} = \boldsymbol{\xi}_{M_0}^{(i)})$. The expectation value of the quantity of interest can be calculated exactly on the coarsest level as

$$\widehat{Y}_0^{\text{exact}} = \widehat{\mathcal{P}}_0^{\text{exact}} = \sum_{i=1}^{n_{\boldsymbol{\xi}}} \mathbb{P}(\boldsymbol{\xi} = \boldsymbol{\xi}^{(i)}) \mathcal{P}_0^{(i)}. \tag{6.3}$$

For the three-point approximation, for example, we need to choose from the 3^d possible values of $\boldsymbol{\xi}_n$ in each of the M_0 time steps, so $n_{\boldsymbol{\xi}} = (3^d)^{M_0}$ is the number of different samples of $\boldsymbol{\xi}$. Since the estimator contains no sampling error, its variance is zero. In Algorithm 1, we can replace $\widehat{Y}_{0,N_0} \mapsto \widehat{Y}_0^{\text{exact}}$ and $\widehat{V}_{0,N_0} \mapsto 0$ in lines 10 and 12. Effectively, this implies that the sum in line 12 only runs from $j = 1$ to L and it is not necessary to evaluate N_0^+.

Naively, the computational complexity of evaluating Eq. (6.3) is given by the product of the number of different samples and the number of time steps, $n_{\boldsymbol{\xi}} \times M_0 = M_0(3^{dM_0})$. However, using a recursive algorithm, the computational complexity can be reduced to the number of nodes in the product-probability tree, which is only $\mathcal{O}(n_{\boldsymbol{\xi}}) = \mathcal{O}(3^{dM_0})$ Nevertheless, this still grows exponentially with the number M_0 of coarse time steps and so Eq. (6.3) is only competitive for small values of M_0 and d. However, exact evaluation can reduce the overall cost of the algorithm dramatically and this is exploited to significant advantage in §6.3.

We now state and prove a modified complexity theorem that allows for additional bias to be introduced between levels, as well as for a different computational cost on the coarsest level.

Let $\widetilde{\mathcal{P}}_\ell$ be the estimator corresponding to $\widehat{\mathcal{P}}_\ell$, but with increments given by Eq. (4.9). For Gaussian increments these estimators are the same, but they are different when we use 3-point or 4-point approximations. Recall that the fine, level ℓ, sample in each of the estimators $\widehat{Y}_{\ell, N_\ell}$ uses increments sampled directly from the 3-point or 4-point distribution, while the coarse, level $\ell - 1$, sample is computed using two consecutive fine increments and formula Eq. (4.9).

Theorem 6.1. *Let us replace Assumption (ii) of Theorem 2.1 by*

(ii) $\mathbb{E}\left[\widehat{Y}_0\right] = \mathbb{E}\left[\widehat{\mathcal{P}}_\ell\right]$ *and* $\left|\mathbb{E}\left[\widehat{\mathcal{P}}_\ell - \widetilde{\mathcal{P}}_\ell\right]\right| \leq c_0 h_\ell^\gamma,$

for some positive constants c_0 and $\gamma > \alpha \geq \frac{1}{2}$. We suppose that all the other assumptions of Theorem 2.1 hold, except that $\mathrm{Cost}_0^{(\mathrm{MLMC})}$ is not necessarily assumed to be bounded by $c_3 N_0 h_0^{-1}$ any longer. Then, there exists a positive constant c_5 such that for any $\epsilon < 1/e$, there are values M_0, L and N_ℓ for which $\widehat{Y}_{\{N_\ell\}}$ from Eq. (2.2) has a MSE $< \epsilon^2$ and a computational complexity $\mathrm{Cost}^{(\mathrm{MLMC})}$ with bound

$$\mathrm{Cost}^{(\mathrm{MLMC})} \leq \mathrm{Cost}_0^{(\mathrm{MLMC})} + c_5 \epsilon^{-2+1/\gamma}. \tag{6.4}$$

Proof. We only require slight modifications in the proof of (Giles, 2008, Theorem 3.1) to prove this result. In particular, it is sufficient to choose

$$L = \left\lceil \frac{\log_2(\sqrt{3} c_1 T^\alpha \epsilon^{-1})}{\alpha} \right\rceil$$

to bound the bias on the finest level. The factor $\sqrt{3}$ appears, since we now have three error contributions, the bias on the finest level, the bias between levels and the sampling error, and since we require each of these contributions to the MSE to be less than $\epsilon^2/3$.

To guarantee that the bias between levels is less than $\epsilon^2/3$, note that due to assumption (ii) we have

$$\left|\sum_{\ell=0}^{L-1} \mathbb{E}\left[\widehat{\mathcal{P}}_\ell - \widetilde{\mathcal{P}}_\ell\right]\right| \leq \sum_{\ell=0}^{L-1} \left|\mathbb{E}\left[\widehat{\mathcal{P}}_\ell - \widetilde{\mathcal{P}}_\ell\right]\right| \leq c_0 h_0^\gamma \sum_{\ell=0}^{L-1} 2^{-\ell\gamma} < \frac{\sqrt{2} c_0}{\sqrt{2} - 1} h_0^\gamma,$$

and so a sufficient condition is $h_0 \leq c_6 \epsilon^{1/\gamma}$ with $c_6 = \left(\frac{\sqrt{3}}{c_0}\left(1 - \frac{1}{\sqrt{2}}\right)\right)^{1/\gamma}$.

Finally, setting

$$N_\ell = \left\lceil \frac{3 c_2 h_0^2}{\sqrt{2} - 1} \epsilon^{-2} 2^{-3\ell/2} \right\rceil$$

and exploiting standard results about geometric series, we get

$$\sum_{\ell=1}^{L} \mathrm{Var}\left[\widehat{Y}_{\ell, N_\ell}\right] \leq \frac{\epsilon^2}{3}\left(\sqrt{2} - 1\right) \sum_{\ell=1}^{L} \frac{c_2 h_\ell^2}{c_2 h_0^2} 2^{3\ell/2} \leq \frac{\epsilon^2}{3}\left(\sqrt{2} - 1\right) \sum_{\ell=1}^{L} 2^{-\ell/2} \leq \frac{\epsilon^2}{3}.$$

The computational cost can then be bounded by

$$\mathrm{Cost}^{(\mathrm{MLMC})} \leq \mathrm{Cost}_0^{(\mathrm{MLMC})} + c_3 \sum_{\ell=1}^{L} N_\ell h_\ell^{-1} \leq \mathrm{Cost}_0^{(\mathrm{MLMC})} + \frac{3 c_2 c_3 h_0}{\sqrt{2} - 1} \epsilon^{-2} \sum_{\ell=1}^{L} 2^{-\ell/2}$$

which leads to the desired bound with $c_5 = 3 c_2 c_3 c_6 \left(\sqrt{2} - 1\right)^{-2}$. (Note that as in (Giles, 2008) this (optimal) choice of N_ℓ is obtained by minimising the cost on levels 1 to L subject to the constraint that the sum of the variances is less than $\epsilon^2/3$.) $\qquad\square$
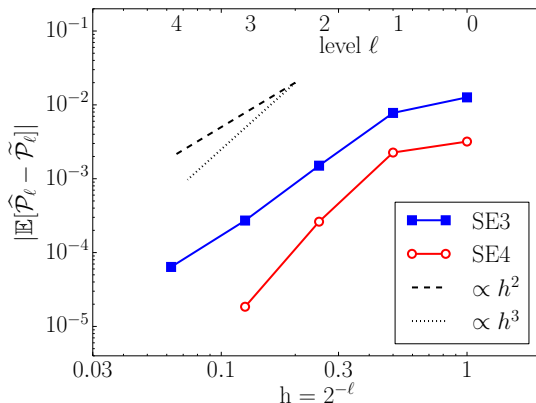
Figure 6: h-dependency of the additional bias term $|\mathbb{E}\left[\widehat{\mathcal{P}}_\ell - \widetilde{\mathcal{P}}_\ell\right]|$ (see (ii) in Theorem 6.1) for $\sigma_0 = \lambda = 1$, $\sigma = 0.4$ computed using the symplectic Euler/exact OU splitting. Results are shown both for three-point (SE3) and four-point (SE4) random variables.

If we use a q-point approximation and the expected value on the coarsest level is computed exactly, as described in §6.2, then $\text{Cost}_0^{(\text{MLMC})} = \mathcal{O}(q^{dM_0}) = \mathcal{O}\left(\eta^{\epsilon^{-1/\gamma}}\right)$, for some $\eta > 1$. Hence, the total cost grows exponentially with ϵ, as expected. However, for practically relevant values of ϵ, the exponential term may not be dominant and we may get significant computational savings, as we will see in the next section. Note that $\gamma = 2$ for the three-point and $\gamma = 3$ for the four-point case, leading to a cost of $\mathcal{O}(\epsilon^{-3/2})$ and $\mathcal{O}(\epsilon^{-5/3})$ for the computation of the correction terms on levels 1 to L, respectively.

Since the sampling of discrete random variables is significantly cheaper, it may also be of interest to use standard Monte Carlo on the coarsest level, as in the earlier sections of this paper. If we slightly increase the constant in the formula for N_ℓ, $\ell = 1, \ldots, L$, in the proof of Theorem 6.1 and choose $N_0 = \mathcal{O}(\epsilon^{-2})$ such that the total variance over all levels is below $\epsilon^2/3$, then the dominant cost will be $\text{Cost}_0^{(\text{MLMC})} = \mathcal{O}(N_0 h_0^{-1})$, and so $\text{Cost}^{(\text{MLMC})} \leq c_5^* \epsilon^{-2-1/\gamma}$, which will be $\mathcal{O}(\epsilon^{-5/2})$ and $\mathcal{O}(\epsilon^{-7/3})$ in the three- and four-point cases, respectively. However, in practice c_5^* is significantly smaller than the constant c_4 in Theorem 2.1, so that for moderate values of ϵ, the use of discrete random variables will pay off.

6.3 Numerical experiments with discrete random variables

We carry out numerical experiments as in §5.1 with the damped harmonic oscillator, but change the parameters slightly to $\omega_0 = \lambda = 1$, $\sigma = 0.4$ (i.e., smaller noise). Instead of sampling from a Gaussian distribution, we use discrete random numbers, which introduce an additional bias as discussed above. To quantify this bias numerically, we plot the difference $\left|\mathbb{E}\left[\widehat{\mathcal{P}}_\ell - \widetilde{\mathcal{P}}_\ell\right]\right|$ in Figure 6 for the symplectic Euler/exact OU method both for three-point (SE3) and four point (SE4) distributions. The figure shows that, as predicted in Kloeden *et al.* (1995), the additional bias is proportional to h^2 for SE3 and to h^3 for SE4. We have also studied the dependence on the noise term (not shown here) and found that, as σ gets smaller, the additional bias is reduced very rapidly (proportional to σ^3 and σ^4, respectively).

For the same setup, we measure the computational cost and the total error (consisting of the statistical error, discretisation error and the additional bias introduced by sampling from discrete distributions). We calculate the same quantity of interest as in §5.1. Figure 7 shows the results both for Gaussian random variables (SEG) and for the three- and four-point distributions (SE3 and SE4) with $M_0 = 8$. For the discrete distributions, the coarse-grid expectation value is calculated exactly. For the three-point distribution, we also varied the number of time steps on the coarsest level and use $M_0 = 4$ (SE3-), $M_0 = 8$ (SE3) and $M_0 = 16$ (SE3+). In each case, we only show results up to the point where the additional bias error becomes too large. For
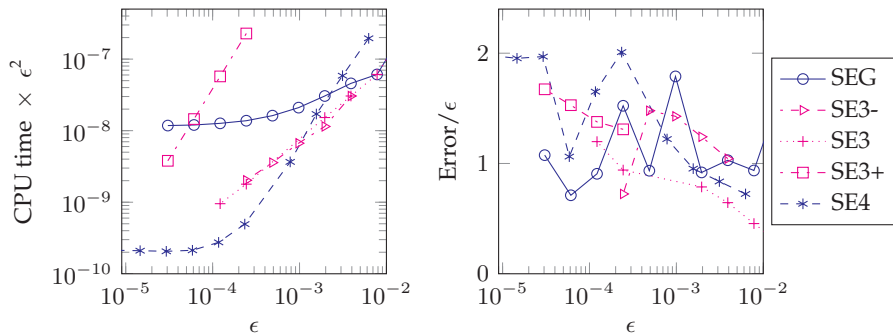
19

Figure 7: Harmonic oscillator with $\lambda = 1$ and $\sigma = 0.4$ computed using the symplectic Euler/exact OU splitting method using three-point (SE3-, SE3, SE3+) and four-point (SE4) random variables.

| | Harmonic Oscillator (Set 2) | | Double-well Potential | |
	$\epsilon = 2.4 \times 10^{-4}$	ratio	$\epsilon = 2.44 \times 10^{-3}$	ratio
MC w. EMG	467 sec	13× slower	1710 sec	378× slower
MLMC w. EMG	33.8 sec	1	45.2 sec	1
MLMC w. SEGe	2.15 sec	15× faster	10.5 sec	4.3× faster

Table 1: Comparison of Monte Carlo with Euler–Maruyama, MLMC with Euler–Maruyama and MLMC with the symplectic Euler/OU integrator and extrapolation (using Gaussian increments).

fixed ϵ, the SE3+ method is more expensive than SE3 and SE3-, since the cost of the exact coarse-level evaluation grows exponentially with the number of time steps. On the other hand, using smaller time steps on the coarsest level allows the use of this method for smaller values ϵ where the additional bias becomes too large for SE3- and SE3. The additional bias in the SE4 method is so small that the method can be used up to values as small as $\epsilon = 10^{-5}$. Comparing the cost of this method to the Gaussian case shows that using a discrete four-point distribution is more than 50-times faster in this case.

We conclude that, if used with caution, approximating the Gaussian increments in Eq. (1.2) by discrete approximations and calculating the coarse-level expectation value exactly can significantly improve the efficiency of the multilevel method.

7 Conclusion

Table 1 summarises our findings: MLMC gives a significant speed-up over the traditional Monte Carlo computation of averages and, even though the optimal complexity estimate $\mathcal{O}(\epsilon^{-2})$ for Monte Carlo-type methods holds for all the integrators under study, there is significant variation between the integrators. Splitting methods are particularly appropriate for the Langevin equation and using the exact OU solution yields a more stable integrator than Euler–Maruyama, even though both integrators are explicit. In the experiments, the difference in computation time between Euler–Maruyama and the splitting methods is greater when the dissipation λ is higher, since Euler–Maruyama suffers from a more severe time-step restriction (cf. Figures 2–5).

This paper also introduced an alternative analysis method for MLMC based on modified equations. It provides a convenient approach to MLMC through weak-approximation theory; strong-approximation theory is only needed to relate the original and modified equations and not the numerical methods. This accommodated the use of the splitting method and the exact OU solution easily.

The weak-approximation analysis motivated the use of discrete random variables, such as three- and four-point approximations to the Gaussian. In an example with small noise ($\sigma = 0.4$ and $\lambda = 1$), we saw between one and two orders of magnitude speed-up for a useful range of

ϵ because we can evaluate the coarse level exactly. This method is easy to implement and it works well because the dominant cost lies on the coarsest level for these problems. While the speed improvements are impressive, this method should be used with care as it introduces an extra bias error. The extra bias can be estimated as shown in Figure 6. The improvement would be less dramatic in higher dimensions as the number of samples required would increase dramatically and it may be impossible to compute the coarse level exactly. As an interesting side result, we proved a modified complexity theorem that allows for extra bias to be introduced between levels in MLMC.

A Proof of Lemma 3.3

Proof. This is a standard Gronwall argument with the Ito isometry. The integral equation for the difference $\boldsymbol{X}(t) - \boldsymbol{X}_h(t)$ is

$$\boldsymbol{X}(t) - \boldsymbol{X}_h(t) = \int_0^t \big(\boldsymbol{f}(\boldsymbol{X}(s)) - \tilde{\boldsymbol{f}}(\boldsymbol{X}_h(s)) \big)\, ds$$
$$+ \int_0^t \Big(G(\boldsymbol{X}(s)) - \widetilde{G}(\boldsymbol{X}_h(s)) \Big)\, d\boldsymbol{W}(s),$$

where $\tilde{\boldsymbol{f}} \coloneqq \boldsymbol{f} + h\boldsymbol{f}_1$ and $\widetilde{G} \coloneqq G + hG_1$. By conditions (i) and (ii),

$$\big\| \boldsymbol{f}(\boldsymbol{X}(s)) - \boldsymbol{f}(\boldsymbol{X}_h(s)) - h\boldsymbol{f}_1(\boldsymbol{X}_h(s)) \big\|_{L^2(\Omega,\mathbb{R}^d)} \leq L \big\| \boldsymbol{X}(s) - \boldsymbol{X}_h(s) \big\|_{L^2(\Omega,\mathbb{R}^d)} + C_1 h$$

and similarly for G. Assume $\boldsymbol{X}(0) = \boldsymbol{X}_h(0)$. The Ito isometry and Jensen's inequality give

$$\mathbb{E}\Big[\big\| \boldsymbol{X}(t) - \boldsymbol{X}_h(t) \big\|^2 \Big] \leq 2t \int_0^t \mathbb{E}\Big[\big\| \boldsymbol{f}(\boldsymbol{X}(s)) - \tilde{\boldsymbol{f}}(\boldsymbol{X}_h(s)) \big\|^2 \Big]\, ds$$
$$+ 2 \int_0^t \Big\| G(\boldsymbol{X}(s)) - \widetilde{G}(\boldsymbol{X}_h(s)) \Big\|_{\mathrm{F}}^2\, ds$$
$$\leq 4t \int_0^t L^2 \mathbb{E}\Big[\big\| \boldsymbol{X}(s) - \boldsymbol{X}_h(s) \big\|^2 \Big] + C_1^2 h^2\, ds$$
$$+ 4 \int_0^t L^2 \mathbb{E}\Big[\big\| \boldsymbol{X}(s) - \boldsymbol{X}_h(s) \big\|^2 \Big] + C_1^2 h^2\, ds.$$

Gronwall's inequality gives $\mathcal{O}(h^2)$ bounds on $\mathbb{E}\big[\| \boldsymbol{X}(t) - \boldsymbol{X}_h(t) \|^2 \big]$. A similar argument can be applied to $\boldsymbol{Y}$ and applying the Lipschitz condition on ψ completes the proof. $\qquad\square$

References

Abdulle, A., & Blumenthal, A. 2013. Stabilized multilevel Monte Carlo method for stiff stochastic differential equations. *Journal of Computational Physics*, **251**, 445–460.

Abdulle, A., Vilmart, G., & Zygalakis, K. C. 2014. Long time accuracy of Lie–Trotter splitting methods for Langevin dynamics. *To appear in SIAM J. Numerical Analysis.*

Anderson, D. F., & Higham, D. J. 2011. Multilevel Monte Carlo for continuous time Markov chains, with applications in biochemical kinetics. *Multiscale Modeling & Simulation*, **10**(1), 146–179.

Barth, A., Schwab, C., & Zollinger, N. 2011. Multi-level Monte Carlo finite element method for elliptic PDEs with stochastic coefficients. *Numerische Mathematik*, **119**(1), 123–161.

Beard, D. A., & Schlick, T. 2000. Inertial stochastic dynamics: I. Long timestep methods for Langevin dynamics. *Journal of Chemical Physics*, **112**(17), 7313–7322.

Belomestny, Denis, & Nagapetyan, Tigran. 2014. Multilevel path simulation for weak approximation schemes. *arXiv:1406.2581*.

Bou-Rabee, Nawaf, & Owhadi, Houman. 2009. Stochastic variational integrators. *IMA J. Numer. Anal.*, **29**(2), 421–443.

Bou-Rabee, Nawaf, & Owhadi, Houman. 2010. Long-run accuracy of variational integrators in the stochastic context. *SIAM J. Numer. Anal.*, **48**(1), 278–297.

Brunge, A., Brooks, C. L., & Karplus, M. 1984. Stochastic boundary conditions for molecular dynamics simulations of ST2 water. *Chem. Phys. Lett.*, **105**, 495–400.

Cliffe, K. A., Giles, M. B., Scheichl, R., & Teckentrup, A. L. 2011. Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients. *Computing and Visualization in Science*, **14**(1), 3–15.

Debussche, A., & Faou, E. 2012. Weak backward error analysis for SDEs. *SIAM J. Numer. Anal.*, **50**(3), 1735–1752.

Dereich, S., & Heidenreich, F. 2011. A multilevel Monte Carlo algorithm for Lévy-driven stochastic differential equations. *Stochastic Processes and their Applications*, **121**, 1565–1587.

Giles, M. B. 2008. Multilevel Monte Carlo path simulation. *Oper. Res.*, **56**(3), 607–617.

Giles, M. B., & Reisinger, C. 2012. Stochastic finite differences and multilevel Monte Carlo for a class of SPDEs in finance. *SIAM Journal of Financial Mathematics*, **3**(1), 572–592.

Giles, M. B., & Szpruch, L. 2013. Multilevel Monte Carlo methods for applications in finance. *Pages 3–47 of:* Gerstner, T., & Kloeden, P. (eds), *Recent Developments in Computational Finance.* World Scientific Press.

Hairer, E., Lubich, C., & Wanner, G. 2010. *Geometric Numerical Integration.* Springer Series in Computational Mathematics, vol. 31. Springer, Heidelberg. Structure-preserving Algorithms for Ordinary Differential Equations, Reprint of the second (2006) edition.

Heinrich, S. 2001. Multilevel Monte Carlo Methods. *Pages 58–67 of: Large-scale scientific computing.* Springer.

Hoel, H., von Schwerin, E., Szepessy, A., & Tempone, R. 2012. Adaptive multilevel Monte Carlo simulation. *Numerical Analysis of Multiscale Computations*, **82**, 217–234.

Hutzenthaler, M., Jentzen, A., & Kloeden, P. E. 2013. Divergence of the multilevel Monte Carlo Euler method for nonlinear stochastic differential equations. *Ann. Appl. Probab.*, **23**(5), 1913–1966.

Kloeden, P. E., & Platen, E. 1992. *Numerical solution of stochastic differential equations.* Applications of Mathematics (New York), vol. 23. Springer-Verlag, Berlin.

Kloeden, P. E., Platen, E., & Hofmann, N. 1995. Extrapolation methods for the weak approximation of Itô diffusions. *SIAM J. Numer. Anal.*, **32**(5), 1519–1534.

Kopec, M. 2013. *Weak backward error analysis for Langevin process.* arXiv:1310.2599.

Leimkuhler, B., & Reich, S. 2004. *Simulating Hamiltonian Dynamics.* Cambridge Monographs on Applied and Computational Mathematics, vol. 14. Cambridge University Press.

Leimkuhler, B., Matthews, Ch., & Stoltz, G. 2013. The computation of averages from equilibrium and non-equilibrium Langevin molecular dynamics. *arXiv preprint 1308.5814*.

Leimkuhler, B., Matthews, C., & Tretyakov, M. V. 2014. On the long-time integration of stochastic gradient systems. *Proc. Roy. Soc. A*, **470**(2170).

Leimkuhler, Benedict, & Matthews, Charles. 2013. Rational Construction of Stochastic Numerical Methods for Molecular Sampling. *Applied Mathematics Research eXpress*, **2013**(1), 34–56.

Lemaire, V., & Pagès, G. 2013. *Multilevel Richardson–Romberg extrapolation.* Preprint arXiv:1401.1177.

Mishra, S., Schwab, C., & Šukys, J. 2012. Multi-level Monte Carlo finite volume methods for nonlinear systems of conservation laws in multi-dimensions. *Journal of Computational Physics*, **231**(8), 3365–3388.

Risken, H. 1996. *The Fokker–Planck Equation: Methods of Solution and Applications.* Lecture Notes in Mathematics. Springer Berlin Heidelberg.

Rodean, H. C. 1996. Stochastic Lagrangian models of turbulent diffusion. *Meteorological Monographs*, **26**(48), 1–84.

Shardlow, T. 2006. Modified equations for stochastic differential equations. *BIT*, **46**(1), 111–125.

Skeel, R. D., & Izaguirre, J. A. 2002. An impulse integrator for Langevin dynamics. *Molecular Physics*, **10**(24).

Talay, D., & Tubaro, L. 1990. Expansion of the global error for numerical schemes solving stochastic differential equations. *Stochastic Anal. Appl.*, **8**(4), 483–509.

Wang, W., & Skeel, R. D. 2003. Analysis of a few numerical integration methods for the Langevin equation. *Molecular Physics*, **101**(14), 2149–2156.

Zygalakis, K. C. 2011. On the existence and the applications of modified equations for stochastic differential equations. *SIAM J. Sci. Comput.*, **33**(1), 102–130.