Journal of Computational Physics 251 (2013) 445–460

Contents lists available at SciVerse ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Stabilized multilevel Monte Carlo method for stiff stochastic differential equations

Assyr Abdulle *, Adrian Blumenthal

ANMC, Mathematics Section, École Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland

ARTICLE INFO

Article history:
Received 3 January 2013
Received in revised form 6 May 2013
Accepted 30 May 2013
Available online 18 June 2013

Keywords:
Multilevel Monte Carlo
Stochastic differential equations
Stability
Stiffness
S-ROCK method

ABSTRACT

A multilevel Monte Carlo (MLMC) method for mean square stable stochastic differential equations with multiple scales is proposed. For such problems, that we call stiff, the performance of MLMC methods based on classical explicit methods deteriorates because of the time step restriction to resolve the fastest scales that prevents to exploit all the levels of the MLMC approach. We show that by switching to explicit stabilized stochastic methods and balancing the stabilization procedure *simultaneously* with the hierarchical sampling strategy of MLMC methods, the computational cost for stiff systems is significantly reduced, while keeping the computational algorithm fully explicit and easy to implement. Numerical experiments on linear and nonlinear stochastic differential equations and on a stochastic partial differential equation illustrate the performance of the stabilized MLMC method and corroborate our theoretical findings.

1. Introduction

For computing expectations of functionals depending on a stochastic process, Monte Carlo (MC) methods are an essential tool. In the context of stochastic differential equations (SDEs), sample paths of the solution are computed by a numerical integrator and the MC approach consists in approximating the expected value of a given functional of the solution by the average of the computed samples. Bias and statistical errors are introduced in such an approximation procedure. The bias of the method is related to the weak order of convergence of the considered numerical integrator, while the statistical error scales as the inverse of the square root of the number of samples and involves the variance of the process. This statistical error is a computational burden for many applications and many strategies to reduce the computational cost of MC method have been proposed. We mention the variance reduction techniques such as as estimators based on control variates or antithetic variates (see e.g., [1]).

A recent approach, originating with Heinrich [2] in the context of numerical quadrature, proposed by Giles [3] for SDEs is the so-called multilevel Monte Carlo (MLMC) method that allows to significantly speed up the classical MC method thanks to hierarchical sampling. The main idea of MLMC methods is to apply the MC method for a nested sequence of stepsizes while balancing the number of samples according to the stepsize. Precisely, consider the square root of the mean square error as a measure of the accuracy, and e.g., the Euler–Maruyama (EM) method [4] as the basic numerical integrator. Then, the computational cost of $\mathcal{O}(\varepsilon^{-3})$ for the MC method is reduced to $\mathcal{O}(\varepsilon^{-2}(\log(\varepsilon))^2)$ for the MLMC method to compute the expectation of functionals with an accuracy of $\mathcal{O}(\varepsilon)$.

However, this computational saving is obtained assuming that the coarsest levels of the MLMC method are *accessible*. But it is well known [5] that for classes of problems, e.g., stochastic partial differential equations discretized by the method of

* Corresponding author. Tel.: +41 21 693 03 11.
E-mail addresses: assyr.abdulle@epfl.ch (A. Abdulle), adrian.blumenthal@epfl.ch (A. Blumenthal).

0021-9991/$ - see front matter © 2013 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.jcp.2013.05.039

lines, stability issues with standard explicit methods can prevent to take coarse stepsizes. Indeed, the wide range of scales present in the SDEs forces the numerical integrator to resolve the fastest scale leading to a possible severe stepsize reduction. In this paper we consider mean square stable stiff systems of SDEs for which standard explicit integrators, e.g., the well-known Euler–Maruyama method, face a severe stepsize restriction [5,6]. Such problems and related computational issues arise in the modeling of many problems in biology, chemistry, physics or economics [7–10]. We call these systems stiff. Another finding of this paper is that even for SDEs usually characterized as *nonstiff* but with significant noise, stepsize restriction prevents to use the EM method for all levels of the MLMC. We emphasize that in the SDE context there exists, besides mean square stable problems, various other classes of interesting problems with multiple scales that need other numerical treatments [11,12].

To the best of the authors knowledge the issue of applying MLMC method for stiff SDEs has not been addressed in the literature. We note however that related work extending the MLMC method for problems with multiple scales in space has recently been proposed in [13] in the context of numerical homogenization of stochastic elliptic multiscale PDEs.

One possible strategy to extend MLMC method for stiff problems is to use drift-implicit numerical methods with favorable mean square stability. When applicable, such methods are a good alternative to the stabilized method proposed in this paper. We note however that for large problems, originating for example from a spatially discretized stochastic partial differential equations with stiff nonlinear terms (e.g., reaction terms), solving the full problem with an implicit method is sometimes very difficult if not impossible. For such problems, decoupling the diffusion operator (solved explicitly with a stabilized method [14–16]) from the reaction terms (stiff problems of small dimension at each spatial node solved implicitly) using a splitting method [17] or a partitioned method [18] is a very efficient strategy. In both approaches [18,17] the use of *explicit stabilized methods* is essential. Furthermore, the computational complexity for drift-implicit numerical methods is somehow less transparent than in the original MLMC method based on explicit integrators as the number of iterations to solve the nonlinear systems has to be accounted for. Precisely balancing the accuracy of the linear solver (e.g., iteration of an (inexact) Newton method) with the level of the MLMC method is a nontrivial task. We refer to [19] for such adaptive inexact Newton methods in the context of deterministic PDEs.

Here we explore another avenue and propose to stabilize the EM method to allow to access the coarse stepsizes of the MLMC method. In turn, our method is as easy to implement as the EM method and switching from such a code to a stabilized one as proposed here for a MLMC implementation is straightforward. For the stabilization procedure, we resort to the S-ROCK methods of weak order one, a class of explicit Chebyshev methods recently introduced for stiff stochastic problems [20,21] and extended for higher weak order in [22]. The stabilized MLMC methods remain fully explicit, as easy to implement as the original MLMC methods based on the EM method but much more efficient as shown in Section 3.2. The S-ROCK methods consist in a family of numerical methods indexed by their stage number. This number can in turn vary to accommodate the required stability requirement. If only one stage is used, the S-ROCK method coincides with the EM method and for nonstiff problems we recover the classical MLMC method. Moreover we would like to point out that in [23] the divergence of the MLMC method using Euler–Maruyama for nonlinear SDEs is discussed.

The paper is organized as follows. In Section 2 the numerical methods used in this paper to approximate stochastic differential equations are described, the order of convergence and the stability are discussed. In Section 3 we discuss the issues faced by the standard MLMC approach in presence of stiffness. We then introduce our stabilized multilevel MC method and discuss its complexity. Numerical experiments on a one-dimensional linear SDE, a two-dimensional nonlinear SDE and a large system of SDEs originating from a SPDE are studied in Section 4 to illustrate the performance of our new MLMC method.

2. Preliminaries

In this paper we consider stochastic processes $(X(t))_{t\in[0,T]}$ on the bounded interval $[0,T]$ described by the stochastic differential equation

$$\begin{cases} \mathrm{d}X(t) = f(X(t))\mathrm{d}t + \sum_{r=1}^{m} g^r(X(t))\mathrm{d}W_r(t), \quad 0 \leqslant t \leqslant T, \\ X(0) = X_0, \end{cases} \tag{2.1}$$

where $X(t)$ is a $\mathbb{R}^d$-valued random variable, $f:\mathbb{R}^d\to\mathbb{R}^d$ is the drift term, $g^r:\mathbb{R}^d\to\mathbb{R}^d$ with $r=1,2,\dots,m$ are the diffusion terms and $W_r(t)$ with $r=1,2,\dots,m$ are independent one-dimensional standard Brownian motions. For simplicity autonomous drift and diffusion functions are considered, but emphasize that a general SDE can always be transformed in such autonomous form. We assume standard Lipschitz and linear growth conditions on the drift and diffusion functions to ensure the existence of a strong solution of the SDE (2.1) (see [24–26]).

To approximate numerically the solution of (2.1) we consider the discrete map

$$X_{n+1} = \Psi(X_n, h, \xi_n), \tag{2.2}$$

where $\Psi(\cdot,h,\xi_n):\mathbb{R}^d\to\mathbb{R}^d$, $X_n\in\mathbb{R}^d$ for $n\geqslant 0$, h denotes the stepsize, and ξ_n denotes a random vector. We recall two concepts of accuracy and stability for the numerical integration of SDEs. A numerical approximation (2.2), starting from the exact initial condition X_0 of (2.1) is said to have strong order of convergence r_s if

$$\exists C \in \mathbb{R}_+ \text{ such that } \max_{0\leqslant n\leqslant T/h} \left(\mathbb{E}\left[|X_n - X(\tau_n)|^2\right]\right)^{1/2} \leqslant Ch^{r_s}, \tag{2.3}$$

where $\tau_n = nh \in [0,T]$ with h small enough and where C is a constant independent of h. A numerical method is said to converge with weak order of convergence r_w if for all functions $\phi : \mathbb{R}^d \to \mathbb{R} \in C_p^{2(\gamma+1)}(\mathbb{R}^d, \mathbb{R})$

$$\exists C \in \mathbb{R}_+ \text{ such that } |\mathbb{E}[\phi(X_n)] - \mathbb{E}[\phi(X(\tau_n)]| \leqslant Ch^{r_w} \tag{2.4}$$

for any $\tau_n = nh \in [0,T]$ fixed and h small enough. Here $C_p^\gamma(\mathbb{R}^d, \mathbb{R})$ denotes the space of γ times continuously differentiable functions $\mathbb{R}^d \to \mathbb{R}$ with all partial derivatives with polynomial growth.

The simplest method to approximate solutions to (2.1) is a generalization of the Euler method for ordinary differential equation (ODEs), the Euler–Maruyama method. Taking a uniform stepsize h, the method is defined by

$$X_{n+1} = X_n + hf(X_n) + \sum_{r=1}^{m} g^r(X_n)\Delta W_{n,r}, \tag{2.5}$$

where $\Delta W_{n,r} \sim \mathcal{N}(0,h)$, $r = 1,\dots m$ are independent Wiener increments. This method has strong order $\frac{1}{2}$ and weak order 1 in general for a system of Itô SDEs [4]. As we will see in Section 2.1, the method (2.5) requires a stepsize restriction when applied to stiff stochastic problems.

Stabilized explicit numerical integrators, that are efficient for stiff problems, are given by the so-called S-ROCK methods. S-ROCK methods are explicit orthogonal Runge–Kutta Chebyshev methods with an extended mean square stability domain (see Section 2.1). These methods have first been introduced for Stratonovich stochastic differential equations in [27,20] and they have been extended to Itô SDEs in [21]. Here we will focus on the latter. In this paper we consider the s-stage Itô S-ROCK method of weak order 1 and strong order $\frac{1}{2}$ (see [21]). For all integer $s \geqslant 2$ we define the s-stage Itô S-ROCK method as follows:

$$\begin{aligned}
K_0 &= X_{n-1}\\
K_1 &= X_{n-1} + h\frac{\omega_1}{\omega_0} f(K_0)\\
K_i &= 2h\omega_1 \frac{T_{i-1}(\omega_0)}{T_i(\omega_0)} f(K_{i-1}) + 2\omega_0 \frac{T_{i-1}(\omega_0)}{T_i(\omega_0)} K_{i-1} - \frac{T_{i-2}(\omega_0)}{T_i(\omega_0)} K_{i-2}, \quad i = 2,3,\dots,s-1,\\
K_s &= 2h\omega_1 \frac{T_{s-1}(\omega_0)}{T_s(\omega_0)} f(K_{s-1}) + 2\omega_0 \frac{T_{s-1}(\omega_0)}{T_s(\omega_0)} K_{s-1} - \frac{T_{s-2}(\omega_0)}{T_s(\omega_0)} K_{s-2} + \sum_{r=1}^{m} g^r(K_{s-1})\Delta W_{n,r},
\end{aligned}$$

where $\omega_0 = 1 + \frac{\eta}{s^2}$, $\omega_1 = \frac{T_s(\omega_0)}{T_s'(\omega_0)}$ and $\Delta W_{n,r} = W_r(\tau_n) - W_r(\tau_{n-1})$ and we set $X_n = K_s$. Note that $(T_i(x))_{i\geqslant 0}$ are the orthogonal Chebyshev polynomials, which are recursively given by

$$T_0(x) = 1, \;\; T_1(x) = x, \;\; T_i(x) = 2xT_{i-1}(x) - T_{i-2}(x) \quad \text{for } i \geqslant 2, \;\; x \in \mathbb{R}.$$

The parameter η is known as the *damping* parameter and is used to enlarge the width of the stability domain in the direction of the noise. The value of η can be chosen to optimize the stability (in the mean square sense) of the method (see [21]). We note that in the absence of noise, the S-ROCK method coincides with the Chebyshev method introduced in [28]. We note also that for $s = 1$ we take the Euler–Maruyama method (2.5) so that the s-stage Itô S-ROCK method is defined for all $s \geqslant 1$.

2.1. Stability of numerical methods

The efficiency of an approximation does not only depend on the order of convergence but also on its stability that is essential to correctly capture the long time behavior of the exact solution.

Definition 2.1. A stochastic process $(X(t))_{t\geqslant 0}$ is said to be mean square stable if and only if

$$\lim_{t\to\infty} \mathbb{E}\left[X(t)^2\right] = 0.$$

To carry out a stability analysis, the one-dimensional scalar linear SDE specified through

$$\begin{cases} \mathrm{d}X(t) = \lambda X(t)\mathrm{d}t + \mu X(t)\mathrm{d}W(t), \quad 0 \leqslant t, \;\; \lambda \in \mathbb{C}, \;\; \mu \in \mathbb{C}, \\ X(0) = 1, \end{cases} \tag{2.6}$$

is widely used in the literature as test problem (see e.g., [25]). This SDE admits an exact solution given by $X(t) = X_0 \exp\{(\lambda - \mu^2/2)t + \mu W(t)\}$ (see [29]) and it can be shown that the exact solution is mean square stable if and only if

$$\mathcal{R}\{\lambda\} + \frac{1}{2}|\mu|^2 < 0,$$

where $\mathcal{R}\{\cdot\}$ denotes the real part of a complex number. This defines the stability domain of the test problem (2.6):

$$\mathcal{S}_{\text{exact}} := \left\{(\lambda,\mu)\in\mathbb{C}^2 | \mathcal{R}\{\lambda\}+\frac{1}{2}|\mu|^2<0\right\}. \tag{2.7}$$

To avoid stability issues, i.e., restrictions on the stepsize, the aim of numerical methods is to cover as much as possible of this stability domain.

Some comments on the linear scalar test equation. We note that the justification of the test equation (2.7) is delicate for multi-dimensional systems. Indeed the extension of the stability analysis of numerical methods for SDEs already for multi-dimensional linear systems $\mathrm{d}X = AX\mathrm{d}t + \sum_{r=1}^{m} B_r X \mathrm{d}W_r(t)$, where A, B_r are $d\times d$ matrices and $\mathrm{d}W_r$ are independent one-dimensional Wiener processes is difficult in general as such systems cannot be simultaneously diagonalized if A and $B_r, r = 1,2,\dots,m$ do not commute. Attempts to study numerical stability on linear systems have been carried out in [30,31] but these studies do not allow for an easy characterization of stability criterion. Another attempt to generalize the linear test equation has been proposed in [32] using the theory of stochastic stabilization and destabilization [33]. Two sets of test equations with $d = m = 2$ and $d = m = 3$ have been considered. It turns out that the stability behavior for the Euler–Maruyama method (or its generalization obtained by using the θ method for the drift term) applied to these more general test equations is essentially captured by the the linear test equation (2.6). Finally we mention that for non normal drift (2.6) can indeed fail to characterize the stability property (at least in the pre-asymptotic regime) of numerical methods [34,32]. This is already the case in the deterministic setting for the test equation $y' = \lambda y$ (see [35, IV.11]).

Definition 2.2. A numerical method is said to be mean square stable if and only if

$$\lim_{n\to\infty}\mathbb{E}\left[X_n^2\right] = 0.$$

For the Euler–Maruyama method we have a stability domain given by

$$\mathcal{S}_{\text{EM}} := \left\{(p,q)\in\mathbb{C}^2 \;|\; |1+p|^2+q^2<1\right\}, \tag{2.8}$$

where $(p,q) = (h\lambda, \sqrt{h}|\mu|)$ (see [25]). Choosing $(\lambda,\mu)\in\mathbb{R}^2$ such that the linear SDE (2.6) is mean square stable, it can be shown that the stepsize h of the EM method has to satisfy

$$\rho_{\text{EM}} h := \frac{|\lambda|^2}{2|\lambda|-|\mu|^2}h < 1 \iff h < \frac{1}{\rho_{\text{EM}}} \tag{2.9}$$

to guarantee stability of the numerical scheme. In particular, for $\mu = 0$ (deterministic case), $\rho_{\text{EM}} = \frac{|\lambda|}{2}$ and $\rho_{\text{EM}}\to\infty$ for $|\mu|^2\to 2|\lambda|$.

To define the stability domain of S-ROCK methods we first consider

$$\mathcal{S}_{\text{SDE},a} = \left\{(p,q)\in[-a,0]\times\mathbb{R} \;|\; |q|\leqslant\sqrt{-2p}\right\},$$

a *portion* of the true stability region.[1] Furthermore, we define

$$a^* = \sup\left\{a>0 \;|\; \mathcal{S}_{\text{SDE},a}\subset\mathcal{S}_{\text{num}}\right\},$$

where $\mathcal{S}_{\text{num}}$ denotes the stability domain of the numerical approximation scheme (see [21]). In [21] it is shown that S-ROCK methods have stability domains with large a^* and that the growth of the portion of the true stability region increases as $a_s^*\approx c_{\text{SR}}s^2$ with $c_{\text{SR}}\geqslant 0.33$, where s is the number of stages of the S-ROCK method.

Remark 2.3. A crucial property for the S-ROCK methods is that a_s^* grows quadratically with the stage number s, whereas the number of function evaluations only increases linearly with s.

Fig. 1 illustrates for instance on the right-hand side the stability region (dark gray) of the S-ROCK method with $s = 10$ stages and damping parameter $\eta = 5.9$. For comparison Fig. 1 shows on the left-hand side the stability region of the Euler–Maruyama scheme (2.8), which is given by the area in dark gray. The interior part (light gray) of the dashed lines represents the stability region of the true solution. We see that the S-ROCK methods cover a significantly larger region than the EM method. By varying s, any portion of the true stability region can be covered [21]. Similar to (2.9), for $(\lambda,\mu)\in\mathbb{R}^2$ such that the test problem (2.6) is stable, by choosing $a^* = |\lambda|$ and the stage number s such that

$$c_{\text{SR}}s^2 := a^*h \quad \text{with } 0.33\leqslant c_{\text{SR}}\leqslant 1.01 \tag{2.10}$$

the S-ROCK method is mean square stable for any stepsize h. It is worth noting that condition (2.10) is independent of the diffusion term μ, and we will define $\rho_{\text{SR}} := |\lambda|$.

[1] When we restrict the stability domain to real parameters (p,q) we will call this stability domain stability region.

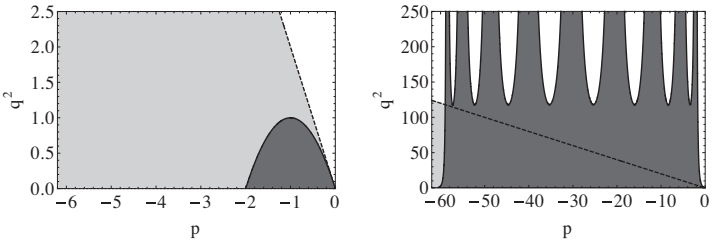

Fig. 1. Stability regions (dark gray) for the Euler–Maruyama method (left-hand side) and the S-ROCK method with stage number $s = 10$ and damping parameter $\eta = 5.9$ (right-hand side). The dashed line delimits the stability region (light gray) of the test problem (2.7).

3. Multilevel Monte Carlo for stiff SDEs

The idea of the multilevel Monte Carlo method [3] is to apply the Monte Carlo method for several nested levels of stepsizes and to compute different numbers of paths on each level, from a few paths when the stepsize is small to many paths when the stepsize is large. By choosing the right balance between the stepsizes and the number of simulated trajectories at each level it is possible to reduce the computational complexity compared to that of the standard Monte Carlo method for a given mean square accuracy.

In the following we use the terms *computational cost* and *computational complexity* synonymously to represent the work of a numerical method defined as the number of function evaluations of a numerical discretization per sample path times the total number of sample paths. This measure of the complexity of numerical algorithms will be used when we compare the performance of various methods.

In this section we discuss the multilevel Monte Carlo method for stiff stochastic differential equations. In the following we first briefly recall the standard MLMC method and show why stability issues restrict this approach for stiff problems. We then present a stabilized multilevel Monte Carlo method using the S-ROCK method.

3.1. Standard multilevel Monte Carlo

Here we present briefly the standard multilevel Monte Carlo approach introduced in [3]. Consider the diffusion process $(X(t))_{t\in[0,T]}$ (with T a fixed positive number) solution of the SDE (2.1) and a Lipschitz continuous function $\phi : \mathbb{R}^d \to \mathbb{R}$. Our aim is to estimate the expectation $\mathbb{E}[\phi(X(T))]$, which we denote by E, from many realizations of the numerical solution of (2.1). Let an integer $k \geqslant 2$ be the refinement factor and let an integer L be the total number of levels. The nested stepsizes of the multilevel Monte Carlo method are given by

$$h_l = \frac{T}{M_l}, \quad l = 0, 1, \ldots, L, \tag{3.1}$$

where $M_l = k^l$ indicates the number of time steps in the discretization over the time interval $[0, T]$ at level l. Let $\phi_l := \phi(X_{M_l}) \approx \phi(X(T))$ be an approximation of $\phi(X(T))$ using a numerical scheme with M_l discretization steps of size h_l. Applying the telescopic sum yields

$$\phi_L = \sum_{l=0}^{L} (\phi_l - \phi_{l-1}) \quad \text{with } \phi_{-1} \equiv 0.$$

The multilevel Monte Carlo estimator is defined by

$$E^* := \sum_{l=0}^{L} E_l^* \quad \text{with } E_l^* := \frac{1}{N_l} \sum_{i=1}^{N_l} \left(\phi_l^{(i)} - \phi_{l-1}^{(i)} \right)$$

a sample average over N_l independent samples. We emphasize that the estimates $\phi_l^{(i)}$ and $\phi_{l-1}^{(i)}$ are based on the same diffusion path, i.e., the same Brownian motion path. The mean square error, a measure of accuracy for estimators (see e.g., [1]), of E^* can be decomposed as

$$\mathrm{MSE}(E^*) = \mathbb{E}\left[(E^* - E)^2 \right] = \mathbb{E}\left[(E^* - \mathbb{E}[E^*])^2 \right] + (\mathbb{E}[E^*] - E)^2 = \mathrm{Var}(E^*) + (\mathrm{bias}(E^*))^2. \tag{3.2}$$

Since the estimates $\phi_l^{(i)}$ are identically distributed, the following holds

$$\mathbb{E}[E^*] = \sum_{l=0}^{L} \mathbb{E}\left[\frac{1}{N_l} \sum_{i=1}^{N_l} \left(\phi_l^{(i)} - \phi_{l-1}^{(i)} \right) \right] = \sum_{l=0}^{L} \mathbb{E}[\phi_l - \phi_{l-1}] = \mathbb{E}[\phi_L].$$

Using this equality and considering a numerical integrator with weak order of convergence 1 yields

$$\mathrm{bias}(E^*) = \mathbb{E}[E^*] - E = \mathbb{E}[\phi_L] - E = \mathcal{O}\left(k^{-L}\right). \tag{3.3}$$

Moreover, we observe that by the Cauchy–Schwarz inequality

$$\mathrm{Var}(\phi_l - \phi_{l-1}) \leqslant \left(\mathrm{Var}(\phi_l - E)^{1/2} + \mathrm{Var}(\phi_{l-1} - E)^{1/2}\right)^2 \tag{3.4}$$

and since ϕ is Lipschitz continuous and a strong order $\frac{1}{2}$ is assumed

$$\mathrm{Var}(\phi_l - E) \leqslant \mathbb{E}\left[(\phi_l - E)^2\right] = \mathbb{E}\left[\left(\phi(X_{M_l}) - \phi(X(T))\right)^2\right] \leqslant Ck^{-l}. \tag{3.5}$$

Therefore, using (3.4) and (3.5) we obtain

$$\mathrm{Var}(E^*) = \sum_{l=0}^{L} \frac{\mathrm{Var}(\phi_l - \phi_{l-1})}{N_l} = C\sum_{l=0}^{L} \frac{k^{-l}}{N_l}, \tag{3.6}$$

where C is a positive constant. Assuming now a mean square accuracy of $\mathrm{MSE}(E^*) = \mathcal{O}(\varepsilon^2)$ and considering (3.2) and (3.3) yields $\varepsilon = k^{-L}$. Inspired by (3.6) the number of simulations per level l is chosen such that $N_l = k^{2L}k^{-l}L$, which guarantees that $\mathrm{Var}(E^*) = \mathcal{O}(\varepsilon^2)$ as L tends to infinity. It is straightforward to show that the corresponding computational complexity of E^* is given by

$$\mathrm{Cost}(E^*) = \sum_{l=0}^{L} N_l M_l (1+m) = \mathcal{O}(\varepsilon^{-2}(\log(\varepsilon))^2),$$

which is a significant improvement over the standard Monte Carlo method with a computational complexity of $\mathcal{O}(\varepsilon^{-3})$. However, one has to be careful when applying the standard MLMC approach for stiff systems as we show in the next section. Indeed, stability of the numerical method used in the standard multilevel Monte Carlo approach is assumed to ensure that all levels of the method are accessible [3]. This will not be the case for stiff problems as will be discussed in the next section.

3.1.1. Multilevel Monte Carlo for stiff SDEs using Euler–Maruyama

Assume a mean square stable problem for which a standard numerical method is only mean square stable for a stepsize smaller than a certain threshold. In such a case the multilevel Monte Carlo method cannot be applied at the levels whose stepsize is larger than this threshold. Inspired by the mean square stable one-dimensional scalar linear SDE (2.6) the following stability constraint is assumed:

$$k^{-l_{\mathrm{EM}}}\rho \leqslant 1, \tag{3.7}$$

where l_{EM} corresponds to the largest possible stepsize $h_{l_{\mathrm{EM}}}$ such that the Euler–Maruyama method is stable for a given stiffness parameter denoted by ρ.

Remark 3.1. For example, for the test problem (2.6), $\rho = \rho_{\mathrm{EM}} = \frac{|\lambda|^2}{2|\lambda|-|\mu|^2}$, and thus, $l_{\mathrm{EM}} = \frac{\log\left(\frac{|\lambda|^2}{2|\lambda|-|\mu|^2}\right)}{\log(k)}$ (see (2.9)). For $l < l_{\mathrm{EM}}$ the EM cannot be applied as the integration is unstable. We emphasize that large l_{EM} can arise in situations usually characterized as nonstiff, i.e., when $|\lambda|$ is small but $|\mu|$ close to $\sqrt{2|\lambda|}$ (see Fig. 4).

Remark 3.2. Note that in (3.7) we assume a relatively small value of T and we willingly ignore T. For a large value of T the following results remain valid by replacing the stiffness parameter ρ by the product $T\rho$. Hence increasing T has the same effect as increasing the stiffness.

Suppose a mean square accuracy of $k^{-2L} = \varepsilon^2$ is desired. We distinguish two cases.

(a) No MLMC: $l_{\mathrm{EM}} > L$.

If l_{EM} is larger than L, then all the stepsize h_l (with $l \in \{0, 1, \ldots, L\}$) are too large to account for stability. Thus, the multilevel Monte Carlo approach cannot be applied and standard Monte Carlo has to be used instead with $M_{l_{\mathrm{EM}}}$ time steps. Therefore, in this case a mean square accuracy of $\mathcal{O}(\varepsilon_{\mathrm{MC}}^2)$ with $\varepsilon_{\mathrm{MC}} = k^{-l_{\mathrm{EM}}}$ is achieved and a computational cost of $\mathcal{O}(\varepsilon_{\mathrm{MC}}^{-3})$ is necessary. We emphasize that $\varepsilon_{\mathrm{MC}} = k^{-l_{\mathrm{EM}}}$ is smaller than the required accuracy $\varepsilon = k^{-L}$ and in turn $\mathcal{O}(\varepsilon_{\mathrm{MC}}^{-3})$ is larger than $\mathcal{O}(\varepsilon^{-3})$.

(b) MLMC: $0 < l_{\mathrm{EM}} \leqslant L$.

If l_{EM} lies between 0 and L, only the levels $l_{\mathrm{EM}}, l_{\mathrm{EM}}+1, \ldots, L$ satisfy the stability constraint (3.7), and thus, the multilevel Monte Carlo estimator using the Euler–Maruyama scheme is defined by

$$\widetilde{E} := \sum_{l=l_{\mathrm{EM}}}^{L} \widetilde{E}_l \quad \text{with} \quad \widetilde{E}_l := \frac{1}{N_l}\sum_{i=1}^{N_l}\left(\phi_l^{(i)} - \phi_{l-1}^{(i)}\right)$$

a sample average over N_l independent samples, where $\phi_{l_{\mathrm{EM}}-1} \equiv 0$. As in (3.2) the mean square error of $\widetilde{E}$ can be divided into bias and variance:

$$\mathrm{MSE}\left(\widetilde{E}\right) = \mathrm{Var}\left(\widetilde{E}\right) + \left(\mathrm{bias}\left(\widetilde{E}\right)\right)^2.$$

Taking into account the weak order of convergence 1 of the Euler–Maruyama scheme (see Section 2), the bias of $\widetilde{E}$ is of order k^{-L}, i.e., $\mathrm{bias}\left(\widetilde{E}\right) = \mathcal{O}\left(k^{-L}\right)$. Using the independence of the samples, ϕ being Lipschitz continuous and strong order of convergence $\frac{1}{2}$ of the Euler–Maruyama method, the variance of $\widetilde{E}$ satisfies

$$\mathrm{Var}\left(\widetilde{E}\right) = \sum_{l=l_{\mathrm{EM}}+1}^{L} \frac{\mathrm{Var}(\phi_l - \phi_{l-1})}{N_l} + \frac{\mathrm{Var}(\phi_{l_{\mathrm{EM}}})}{N_{l_{\mathrm{EM}}}} = C\sum_{l=l_{\mathrm{EM}}+1}^{L} \frac{k^{-l}}{N_l} + \frac{\mathrm{Var}(\phi_{l_{\mathrm{EM}}})}{N_{l_{\mathrm{EM}}}}, \tag{3.8}$$

where C is a positive constant. Recall that a mean square accuracy of $k^{-2L} = \varepsilon^2$ is wanted. Inspired by (3.8), the number of simulations per level is chosen such that

$$N_l = \begin{cases} k^{2L}k^{-l}(L - (l_{\mathrm{EM}}+1)) & \text{if } l \in \{l_{\mathrm{EM}}+1, l_{\mathrm{EM}}+2, \ldots, L\}, \\ k^{2L} & \text{if } l = l_{\mathrm{EM}}. \end{cases} \tag{3.9}$$

Hence, for the variance of our estimator $\mathrm{Var}\left(\widetilde{E}\right) = \mathcal{O}(\varepsilon^2)$ holds and the mean square error is indeed $\mathrm{MSE}\left(\widetilde{E}\right) = \mathcal{O}(\varepsilon^2)$. We compute now the computational complexity that is necessary to achieve such a mean square accuracy. Taking the choice of N_l in (3.9) into consideration, we obtain a computational complexity of

$$\begin{aligned}
\mathrm{Cost}\left(\widetilde{E}\right) &= \sum_{l=l_{\mathrm{EM}}+1}^{L} N_l M_l(1+m) + N_{l_{\mathrm{EM}}} M_{l_{\mathrm{EM}}}(1+m) \\
&= \sum_{l=l_{\mathrm{EM}}+1}^{L} k^{2L-l}(L-(l_{\mathrm{EM}}+1))k^l(1+m) + k^{2L}k^{l_{\mathrm{EM}}}(1+m) \\
&= k^{2L}(1+m)\left[(L-(l_{\mathrm{EM}}+1))(L-l_{\mathrm{EM}}) + k^{l_{\mathrm{EM}}}\right] \\
&= \varepsilon^{-2}(1+m)\left[\left(\frac{\log\left(\varepsilon^{l_{\mathrm{EM}}/L}\right)-\log(\varepsilon)}{\log(k)} - 1\right)\left(\frac{\log\left(\varepsilon^{l_{\mathrm{EM}}/L}\right)-\log(\varepsilon)}{\log(k)}\right) + \varepsilon^{-l_{\mathrm{EM}}/L}\right] \\
&\leqslant \varepsilon^{-2}(1+m)\left[C\left(\frac{\log(\varepsilon)}{\log(k)}\right)^2 + \varepsilon^{-l_{\mathrm{EM}}/L}\right] = \mathcal{O}\left(\varepsilon^{-2}\left((\log(\varepsilon))^2 + \varepsilon^{-l_{\mathrm{EM}}/L}\right)\right),
\end{aligned}$$

where C is a positive constant.

Remark 3.3. Note that as l_{EM} tends to L, the computational cost of $\widetilde{E}$ tends to $\mathcal{O}(\varepsilon^{-3})$, the computational cost of the standard Monte Carlo approach. If l_{EM} tends to zero, the computational cost tends to $\mathcal{O}\left(\varepsilon^{-2}(\log(\varepsilon))^2\right)$, which is the computational cost of the multilevel Monte Carlo method for nonstiff SDEs. Indeed in that case, there is no stepsize restriction for the EM method.

3.2. Stabilized multilevel Monte Carlo

We describe now a stabilized multilevel Monte Carlo method, which enables us to use all the levels of the MLMC approach even in presence of stiffness. As numerical integrator we use the S-ROCK method presented in Section 2. The following stability constraint is taken into account: (for $s_l \geqslant 2$)

$$\frac{k^{-l}\rho}{c_{\mathrm{SR}}s_l^2} \leqslant 1, \tag{3.10}$$

where the stiffness parameter ρ and c_{SR} are two positive constants. For the test problem (2.6), $\rho = \rho_{\mathrm{SR}} = |\lambda|$. In other words, the number of stages at level l satisfies $s_l \geqslant \max\left(\sqrt{\frac{\rho}{c_{\mathrm{SR}}}}k^{-l/2}, 2\right)$. For the same reasons as in Remark 3.2 a $\mathcal{O}(1)$ value for T is assumed.

Remark 3.4. The value of c_{SR} depends on s_l, but it can be estimated numerically for any $s_l \geqslant 2$ (see [21]). Starting from about 1 for low value of s, it quickly settles to 0.33 for larger value of s.

Using the same notation as above, the stabilized multilevel Monte Carlo estimator is given by

$$\widehat{E} := \sum_{l=0}^{L} \widehat{E}_l \quad \text{with} \quad \widehat{E}_l := \frac{1}{N_l}\sum_{i=1}^{N_l}\left(\phi_l^{(i)} - \phi_{l-1}^{(i)}\right)$$

a sample average over N_l independent samples. Again we emphasize that the estimates $\phi_l^{(i)}$ and $\phi_{l-1}^{(i)}$ are based on the same Brownian motion path. The mean square error of the stabilized estimator $\widehat{E}$ can be decomposed as in (3.2)

$$\mathrm{MSE}\left(\widehat{E}\right) = \mathrm{Var}\left(\widehat{E}\right) + \left(\mathrm{bias}\left(\widehat{E}\right)\right)^2. \tag{3.11}$$

By the weak order of convergence 1 of the S-ROCK scheme (see Section 2), for the bias the following holds:

$$\mathrm{bias}\left(\widehat{E}\right) = \mathbb{E}\left[\widehat{E}\right] - E = \mathbb{E}[\phi_L] - E = \mathcal{O}\left(k^{-L}\right).$$

For the variance of $\widehat{E}$ we obtain

$$\mathrm{Var}\left(\widehat{E}\right) = \sum_{l=0}^{L} \frac{\mathrm{Var}(\phi_l - \phi_{l-1})}{N_l} = C\sum_{l=0}^{L} \frac{k^{-l}}{N_l} \tag{3.12}$$

with C a positive constant. To establish this we have used the independence of the samples, ϕ being Lipschitz continuous and strong order of convergence $\frac{1}{2}$ of the S-ROCK method (see Section 2). Suppose now that a mean square accuracy of $\mathrm{MSE}\left(\widehat{E}\right) = \mathcal{O}(\varepsilon^2)$ with $\varepsilon = k^{-L}$ is desired. Inspired by (3.12), we set the number of simulations per level l to $N_l = k^{2L}k^{-l}L$ such that $\mathrm{Var}\left(\widehat{E}\right) = Ck^{-2L}(1+\frac{1}{L})$, and thus, $\mathrm{MSE}\left(\widehat{E}\right) = \mathcal{O}(\varepsilon^2)$. The computational complexity to achieve such a mean square accuracy is given by

$$\begin{aligned}
\mathrm{Cost}\left(\widehat{E}\right) &= \sum_{l=0}^{L} N_l M_l (s_l + m) = k^{2L} L\left(\sqrt{\frac{\rho}{c_{\mathrm{SR}}}}\sum_{l=0}^{L} k^{-l/2} + m(L+1)\right) \\
&= k^{2L} L\left(\sqrt{\frac{\rho}{c_{\mathrm{SR}}}}\frac{\sqrt{k} - k^{-L/2}}{\sqrt{k}-1} + m(L+1)\right) \\
&= \varepsilon^{-2}\left(-\frac{\log(\varepsilon)}{\log(k)}\right)\left(\sqrt{\frac{\rho}{c_{\mathrm{SR}}}}\frac{\sqrt{k} - \varepsilon^{1/2}}{\sqrt{k}-1} + m\left(-\frac{\log(\varepsilon)}{\log(k)} + 1\right)\right) \\
&\leqslant C\varepsilon^{-2}\left(|\log(\varepsilon)|\sqrt{\rho} + m(\log(\varepsilon))^2\right) = \mathcal{O}\left(\varepsilon^{-2}(\log(\varepsilon))^2\left(1 + \frac{\sqrt{\rho}}{|\log(\varepsilon)|}\right)\right),
\end{aligned} \tag{3.13}$$

where C is a positive constant. Note that we recover the result for nonstiff problems up to a factor $\sqrt{\rho}$. It is also worth noting that using MLMC with Euler–Maruyama for stiff SDEs only standard Monte Carlo can be applied in the case $l_{\mathrm{EM}} > L$, see Section 3.1.1. The resulting computational complexity is given by $\mathcal{O}(\varepsilon_{\mathrm{MC}}^{-3})$. Taking into account that $\varepsilon = k^{-L} > k^{-l_{\mathrm{EM}}} = \varepsilon_{\mathrm{MC}}$, one observes that the computational cost for stabilized MLMC is significantly smaller.

Remark 3.5. If $l_{\mathrm{EM}} \leqslant L$, then the Euler–Maruyama method can be applied from level l_{EM} up to level L. The variance (3.12) can be decomposed as

$$\mathrm{Var}\left(\widehat{E}\right) = \sum_{l=0}^{l_{\mathrm{EM}}-1} \frac{\mathrm{Var}(\phi_l - \phi_{l-1})}{N_l} + \sum_{l=l_{\mathrm{EM}}}^{L} \frac{\mathrm{Var}(\phi_l - \phi_{l-1})}{N_l}.$$

Using the strong convergence of order 1/2 of the numerical schemes yields $\mathrm{Var}\left(\widehat{E}\right) = C\left(\sum_{l=0}^{l_{\mathrm{EM}}-1}\frac{k^{-l}}{N_l} + \sum_{l=l_{\mathrm{EM}}}^{L}\frac{k^{-l}}{N_l}\right)$. Inspired by this decomposition, the number of simulations per level is chosen according to $N_l = k^{2L}k^{-l}(l_{\mathrm{EM}}-1)$ for $l \in \{0,1,\ldots,l_{\mathrm{EM}}-1\}$ and $N_l = k^{2L}k^{-l}(L - l_{\mathrm{EM}})$ for $l \in \{l_{\mathrm{EM}}, l_{\mathrm{EM}}+1,\ldots,L\}$ such that $\mathrm{MSE}\left(\widehat{E}\right) = \mathcal{O}(\varepsilon^2)$ with $\varepsilon = k^{-L}$. The resulting computational cost is similar.

3.2.1. Stabilized multilevel Monte Carlo versus stabilized single-level Monte Carlo

In the previous section we have seen that the multilevel Monte Carlo method with S-ROCK as numerical integrator requires a computational cost of $\mathrm{Cost}\left(\widehat{E}\right)$, as specified in (3.13), to achieve a mean square accuracy of $\mathrm{MSE}\left(\widehat{E}\right) = \mathcal{O}(\varepsilon^2)$. Using the same numerical method and the same mean square accuracy, the standard Monte Carlo method satisfies $M_L = \mathcal{O}(\varepsilon^{-1}), N_L = \mathcal{O}(\varepsilon^{-2})$ and $s_L = \mathcal{O}(\sqrt{\varepsilon\rho})$ (due to the stability criterion (3.10)), and thus, the computational cost required is given by

$$\mathrm{Cost}\left(\widehat{E}_{\mathrm{MC}}\right) = M_L N_L (s_L + m) = \mathcal{O}\left(\varepsilon^{-5/2}\sqrt{\rho} + \varepsilon^{-3}\right). \tag{3.14}$$

Remark 3.6. In applications ε corresponds to the user's desired accuracy. As in the multilevel construction, Monte Carlo using S-ROCK can be applied for any ε, whereas Monte Carlo using Euler–Maruyama can be subject to stepsize restriction, and thus, one is forced to choose an $\bar{\varepsilon}$ which is significantly smaller than the user's desired accuracy ε.

In Fig. 2 we compare the computational cost of the stabilized MLMC method and the standard MC method using S-ROCK as a basic integrator against the finest stepsize h_L for $k = 2, m = 1$ and different values of the stiffness parameter ρ with $\rho \in \{1, 1000\}$. Recall that $\varepsilon = k^{-L} = h_L$, and thus, as h_L decreases the accuracy increases. One observes that for any stiffness ρ, as h_L decreases the stabilized multilevel Monte Carlo method prevails over the Monte Carlo method based on S-ROCK. For instance, in Fig. 2(b) for $\rho = 1000$, at $h_L = 2^{-20}$ the computational cost of Monte Carlo is about 10^3 times larger than the computational cost of multilevel Monte Carlo.

Note that as the stiffness ρ increases, the number of stages per level s_l increases, and thus, the computational complexity. Since the standard Monte Carlo method only uses s_L stages, whereas the MLMC method uses at each level l s_l stages, the number of function evaluations for standard MC is smaller than for MLMC for small values of L. However, as L increases, the MLMC approach significantly reduces the computational cost compared to the MC approach.

4. Numerical examples

In this section we study the multilevel Monte Carlo method for stiff stochastic differential equations numerically. Comparisons of the MLMC method for SDEs using S-ROCK and Euler–Maruyama, respectively, are carried out first on a one-dimensional linear SDE, followed by a two-dimensional nonlinear SDE and finally on a stochastic partial differential equation. In the following we use a refinement factor of $k = 2$.

4.1. Linear SDE

The first problem taken into account is the scalar linear test problem (2.6) with $t \in [0, 1]$. To test numerically how well the stabilized MLMC method using S-ROCK performs compared to the MLMC method using Euler–Maruyama, we count the number of function evaluations (adding the number of drift and diffusion evaluations) using a total number of levels L, where $L \in \{1, 2, \ldots, 15\}$. We consider two scenarios:

- A setting usually considered as nonstiff where $\lambda = -1$ and $\mu = \sqrt{-2\lambda - \delta}$ with $0 < \delta \leqslant 2$. In this scenario, as $\delta \to 0$ the parameters of the exact SDE are approaching the boundary of the exact stability domain. Then the gap between the EM mean square stability domain and the boundary of the true stability domain (see Fig. 3 left) triggers an increasingly severe stepsize restriction. In turn only limited levels of the MLMC are accessible. In contrast the stabilized MLMC is always applicable and the mean square stability region for large value of $|\mu|$ is much larger and moreover not vanishing with increasing value of $|\mu|$ belonging to the true mean square stability region (see Fig. 3 right).

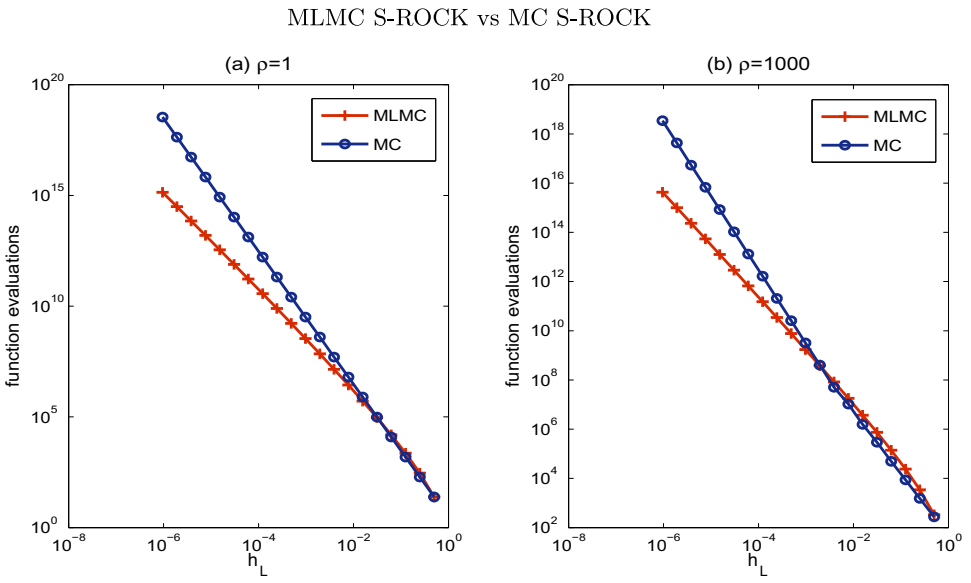

Fig. 2. Computational cost of the stabilized multilevel Monte Carlo and the standard Monte Carlo method (using S-ROCK), respectively, against the finest stepsize h_L for different values of the stiffness parameter ρ.

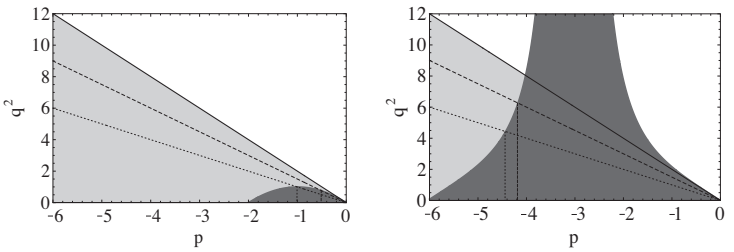

Fig. 3. One-dimensional linear SDE: Stability regions of the test problem (2.6) (light gray), the EM method (dark gray, left-hand side) and the S-ROCK method with $s = 2$ (dark gray, right-hand side). Straight lines of slope -1.5 (dashed) and -1 (dotted), respectively.

- A setting considered as stiff with $\lambda \in \{-1, -100, -10000\}$ and $\mu = \sqrt{|\lambda|}$ where as expected, a decreasing number of levels are accessible for the MLMC based on the EM method in contrast to the stabilized MLMC.

In Fig. 4(a)–(c) we report the results for the first scenario and monitor the number of function evaluations required for the stabilized MLMC method and the MLMC method using EM, respectively, against the finest stepsize h_L. The diffusion is chosen such that $\delta = 0.1$ (Fig. 4(a)), $\delta = 0.01$ (Fig. 4(b)) and $\delta = 0.0001$ (Fig. 4(c)). As expected, the stabilized MLMC method prevails over the MLMC method using Euler–Maruyama. The latter is subject to a stepsize restriction which becomes more severe for decreasing δ.

In Fig. 4(d)–(f) we report the results for the second scenario comparing the the stabilized MLMC method and the MLMC method using EM. We consider a varying drift term with $\lambda \in \{-1, -100, -10000\}$ and a diffusion term given by $\mu = |\lambda|^{1/2}$. In all cases the parameters (λ, μ) lie in the stability region of the test problem (2.7). One observes that as $|\lambda|$ increases, the EM approach can only be used from a certain stepsize on, whereas the S-ROCK approach can be used for any stepsize.

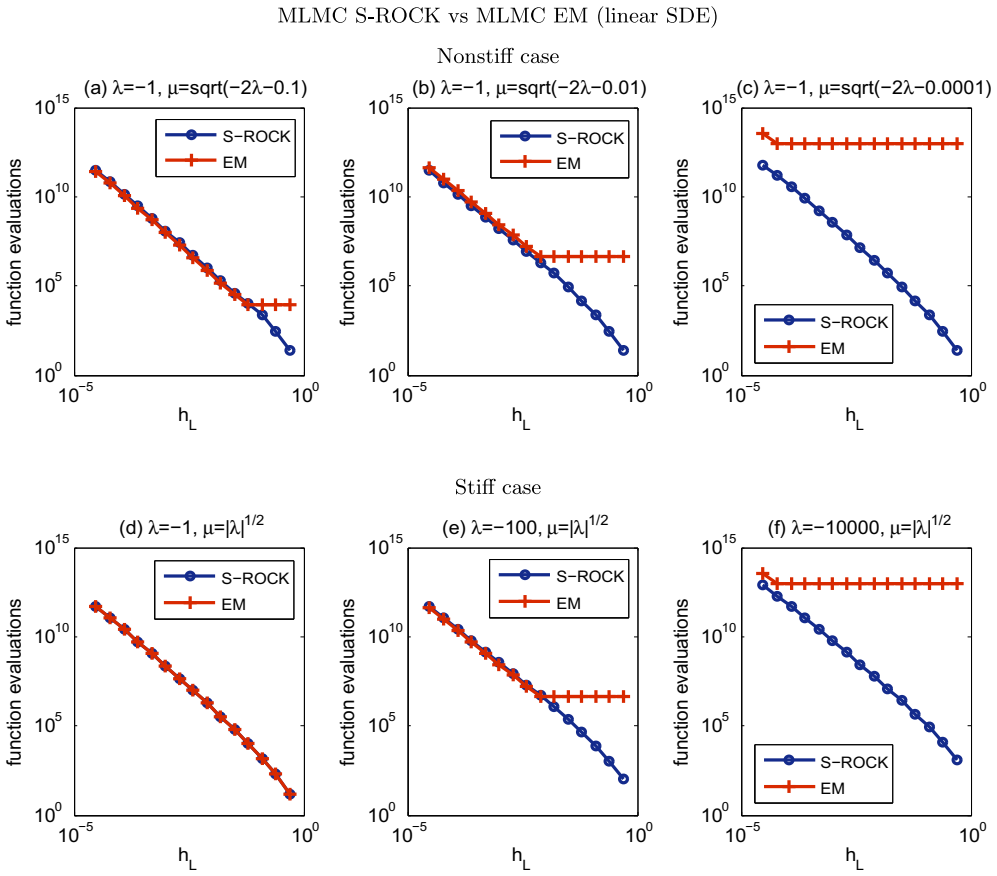

Fig. 4. One-dimensional linear SDE: Function evaluations against finest stepsize h_L comparing the MLMC method using S-ROCK with the MLMC method using Euler–Maruyama.

4.2. Nonlinear SDE

The second stiff numerical experiment that we consider here is a two-dimensional noncommutative stiff SDE given by

$$\begin{cases} \mathrm{d}\begin{pmatrix} X_1(t) \\ X_2(t) \end{pmatrix} = \begin{pmatrix} \alpha(X_2(t)-1) - \lambda_1 X_1(t)(1-X_1(t)) \\ -\lambda_2 X_2(t)(1-X_2(t)) \end{pmatrix} \mathrm{d}t + \begin{pmatrix} -\mu_1 X_1(t)(1-X_1(t)) \\ -\mu_2 X_2(t)(1-X_2(t)) \end{pmatrix} \mathrm{d}W_1(t) + \begin{pmatrix} -\mu_2(1-X_1(t)) \\ 0 \end{pmatrix} \mathrm{d}W_2(t), \quad 0 \leqslant t \leqslant T, \\ \begin{pmatrix} X_1(0) \\ X_2(0) \end{pmatrix} \text{ given}, \end{cases} \tag{4.1}$$

where $(W_1(t))_{t\in[0,T]}$ and $(W_2(t))_{t\in[0,T]}$ represent two independent standard Brownian motions. This model is inspired by the one-dimensional population dynamic model (see [36]). One can observe that $(X_1(t), X_2(t)) = (1,1)\ \forall t \in [0,T]$ represents a stationary solution of (4.1). We carry out a similar numerical experiment as in Section 4.1 by comparing the MLMC method using S-ROCK and Euler–Maruyama, respectively. As parameter we choose $T = 1$, $L \in \{1,2,\ldots,10\}$, $\alpha = 2$, $\lambda_2 = -1$, $\mu_2 = 0.5$ with (λ_1, μ_1) the same as (λ, μ) in the previous section. As initial condition we pick $(X_1(0), X_2(0)) = (0.95, 0.95)$. Note that the two sets of parameters (λ_1, μ_1) and (λ_2, μ_2) both lie in the stability domain with (λ_1, μ_1) governing the stiffness of the SDE.

Fig. 5 illustrates the number of function evaluations against the finest stepsize for the two-dimensional nonlinear noncommutative SDE given in (4.1). The results are similar to the ones of the scalar linear SDE. Note that stability of the approximations has been checked by looking at the second moment at the time end point $T = 1$. The S-ROCK approach can be applied under any choice of the finest stepsize h_L, whereas the Euler–Maruyama approach has some severe stepsize restrictions. Again this corroborates our theoretical findings and illustrates the significant improvement of the stabilized MLMC over the standard MLMC method.

We next study the error behavior of the multilevel Monte Carlo method using S-ROCK applied to the two-dimensional noncommutative nonlinear SDE (4.1). We focus again on the second moment of the stochastic process $\begin{pmatrix} X_1(t) \\ X_2(t) \end{pmatrix}$ at the time

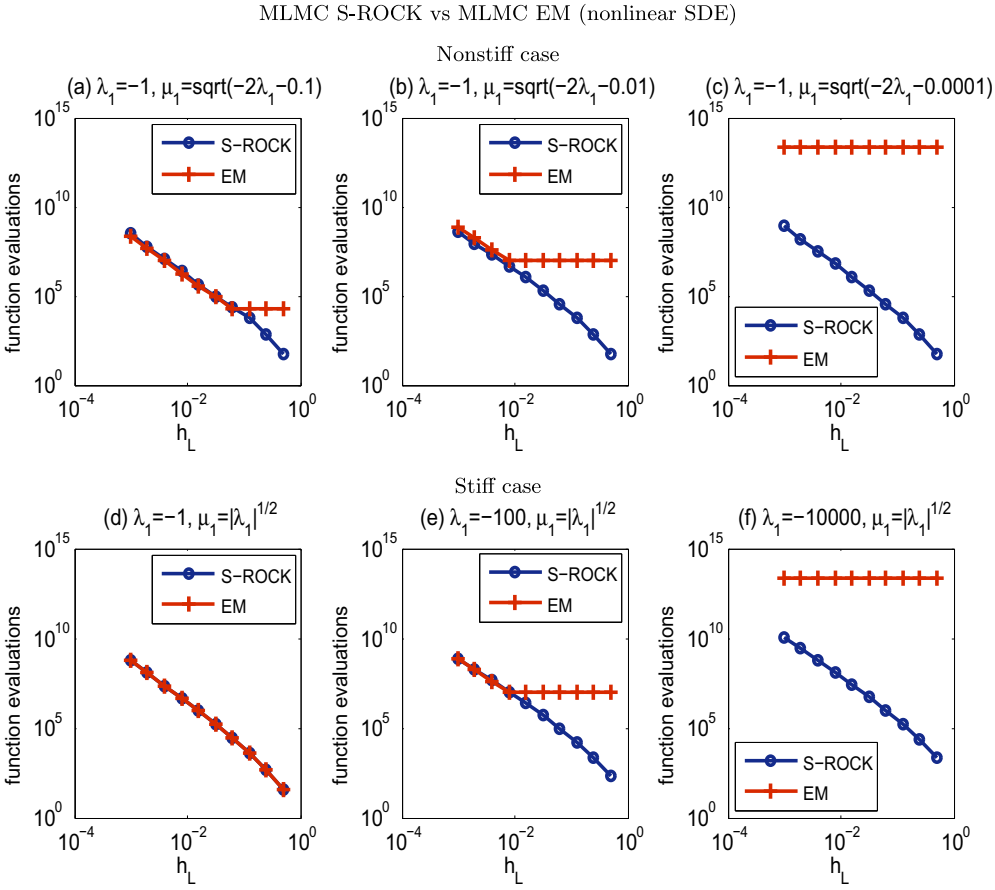

Fig. 5. Two-dimensional nonlinear noncommutative stiff SDE: Function evaluations against finest stepsize h_L comparing the MLMC method using S-ROCK with the MLMC method using Euler–Maruyama for different values of λ_1 and μ_1.

end point $T = 1$. Since an exact solution of the second moment is not known, a reference solution is computed using standard Monte Carlo with Euler–Maruyama and a stepsize of $h = 2^{-12}$. In total 2^{24} Monte Carlo simulations are carried out. Fig. 6 illustrates an approximation of the root mean square error of the second moment of $X_1(t)$ and $X_2(t)$, respectively, at $t = 1$ against the finest stepsize h_L, approximating the expectation by taking a sample average (over 10 samples). We take into account a nonstiff problem with $(\lambda_1 = -1, \mu_1 = \sqrt{-2\lambda_1 - 0.01})$ (see Fig. 6(a)) and a stiff problem with $(\lambda_1 = -100, \mu_1 = |\lambda_1|^{1/2})$ (see Fig. 6(b)). The other parameters have been chosen as above. One observes that in both cases, stiff or nonstiff, the behavior of the RMSE is as expected roughly linear and of slope 1.

4.3. Space-discretized stochastic partial differential equation

The last problem we consider is a stochastic partial differential equation (SPDE) obtained by adding multiplicative noise to the heat equation. The SPDE is specified through

$$\begin{cases} \frac{\partial u(t,x)}{\partial t} = \frac{\partial^2 u(t,x)}{\partial x^2} + \sigma u(t,x)\dot{W}(t,x), \quad (t,x) \in [0,T] \times [0,1], \\ u(0,x) = 1, \quad x \in [0,1], \\ u(t,0) = 5, \quad t \in [0,T], \ \frac{\partial u(t,1)}{\partial x} = 0, \ t \in [0,T], \end{cases} \tag{4.2}$$

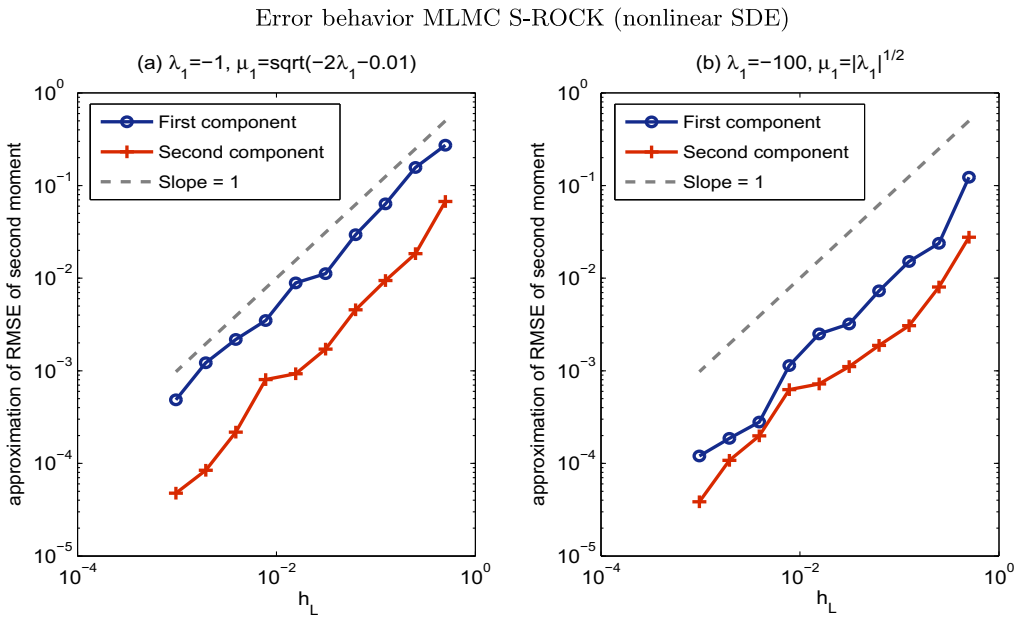

Fig. 6. Two-dimensional nonlinear noncommutative stiff SDE: Error behavior of the MLMC method using S-ROCK applied to (4.1) in a nonstiff (a) and a stiff (b) context. The first and second component correspond to the RMSE of the second moment of $X_1(1)$ and $X_2(1)$, respectively.

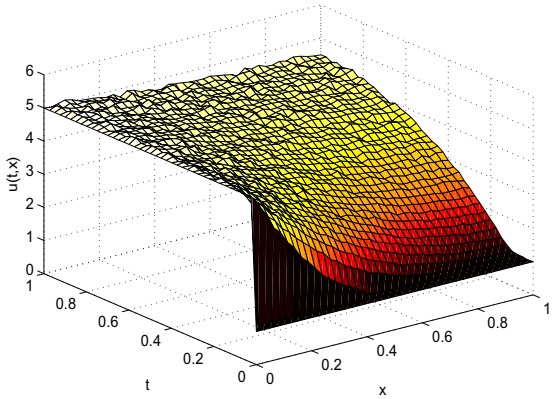

Fig. 7. Stochastic partial differential equation: Numerical approximation of the heat equation with multiplicative noise (4.2) using $\Delta x = 1/40$ and $\Delta t = 1/40$.

where $\dot{W}$ is a space–time white noise and σ a noise parameter. Discretizing in space by using the method of lines yields

$$du_i = \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2} dt + \sigma \frac{u_i}{\sqrt{\Delta x}} dW_i, \quad i = 1, 2, \ldots, \frac{1}{\Delta x} = M,$$

where $u_i \approx u(t, x_i)$ with $x_i = i\Delta x$. By the boundary conditions we have $u_0 = 5$ and $u_{M+1} = u_M$. Note that $W_1, W_2, \ldots, W_M$ are M independent standard Brownian motions and Itô noise has been considered. In the following we use $T = 1$ and $\sigma = 10^{-2}$. Fig. 7 shows one trajectory of the heat equation with noise (4.2) using a space stepsize of $\Delta x = 1/40$ and a time stepsize of $\Delta t = 1/40$.

Remark 4.1. In the following we only vary the time stepsize with the level in the multilevel Monte Carlo construction and consider a fixed spatial discretization. Note that for some applications (especially for multi-dimensional problems) the spatial meshing is not trivial and the flexibility to adapt the spatial mesh is limited (see e.g., [18,17]). We however mention MLMC approaches for SPDEs, where both the time and the space discretizations are adapted [37,38]. These are certainly interesting approaches when applicable but will not be pursued here.

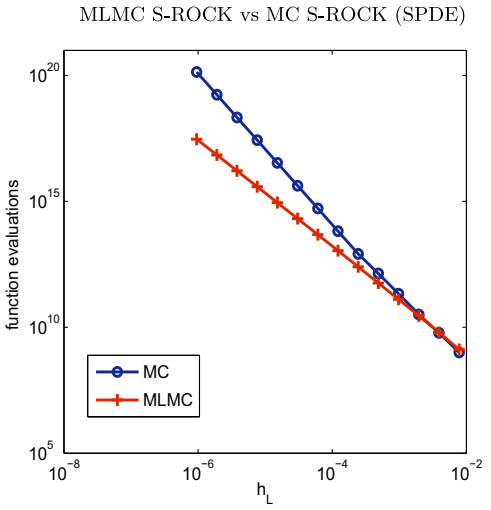

Fig. 8. Stochastic partial differential equation: Number of function evaluations for the stabilized multilevel Monte Carlo method and the stabilized single-level Monte Carlo method, respectively, using $\Delta x = 1/40$ and $h_L \in \{2^{-7}, 2^{-8}, \ldots, 2^{-20}\}$.

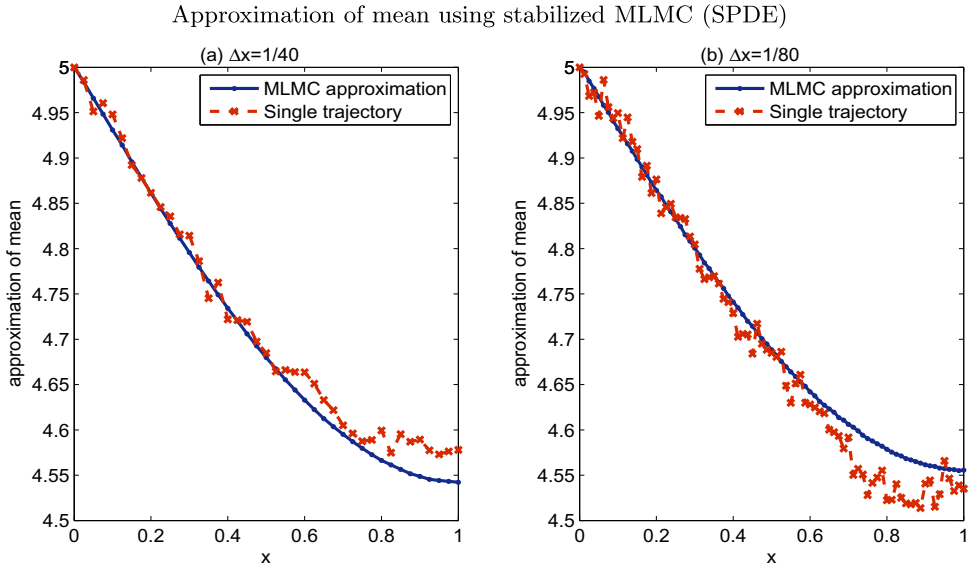

Fig. 9. Stochastic partial differential equation: Approximation of $\mathbb{E}[u(1, x_i)]$, the mean of $u(t, x)$ at $t = 1$, using stabilized MLMC with $L = 10$, $\Delta x = 1/40$ (see (a)) and $\Delta x = 1/80$ (see (b)), respectively. The dotted lines represent a single trajectory at $t = 1$ using S-ROCK with $h = 2^{-10}$.

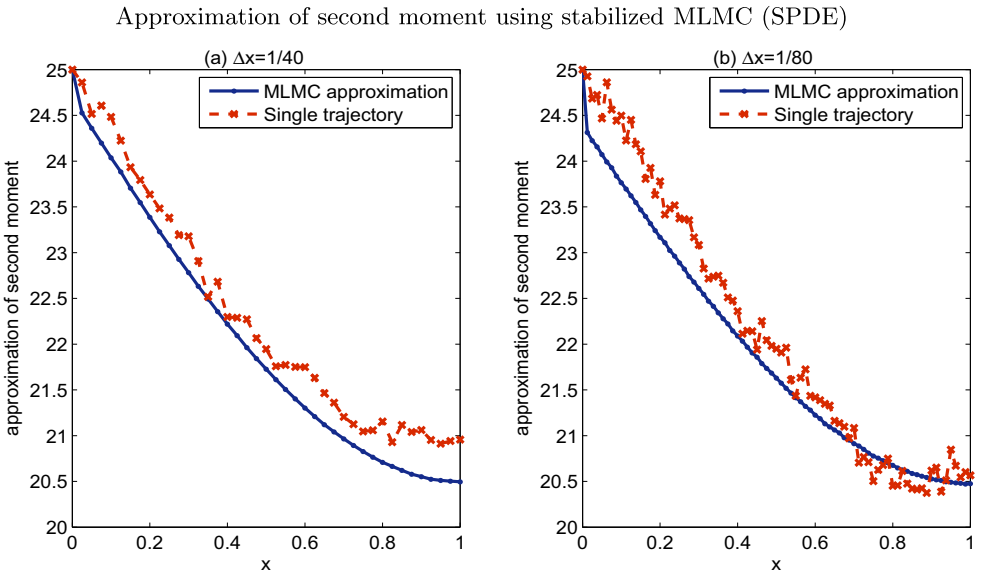

Fig. 10. Stochastic partial differential equation: Approximation of $\mathbb{E}\left[u(1,x_i)^2\right]$, the second moment of $u(t,x)$ at $t=1$, using stabilized MLMC with $L=10$, $\Delta x=1/40$ (see (a)) and $\Delta x=1/80$ (see (b)), respectively. The dotted lines represent a single trajectory at $t=1$ using S-ROCK with $h=2^{-10}$.

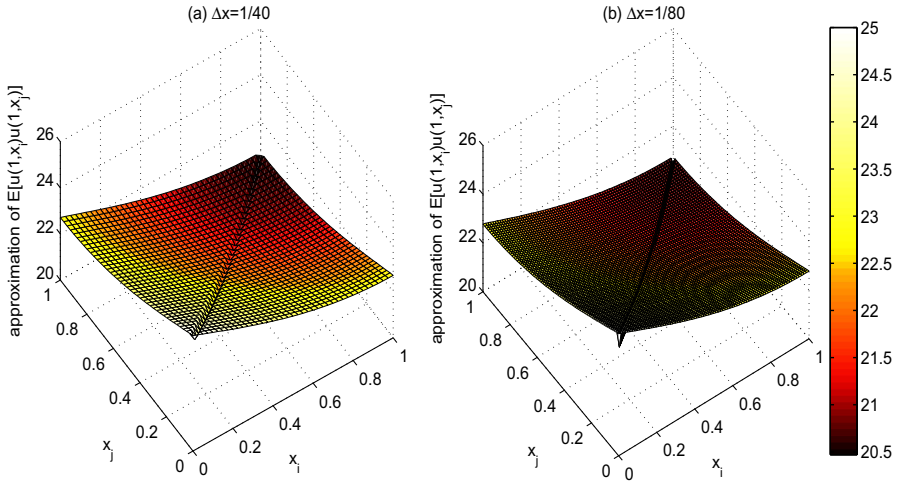

Fig. 11. Stochastic partial differential equation: Approximation of $\mathbb{E}\left[u(t,x_i)u(t,x_j)\right]$ at $t=1$ using stabilized MLMC with $L=10$, $\Delta x=1/40$ (see (a)) and $\Delta x=1/80$ (see (b)).

Fig. 8 illustrates the number of function evaluations of the stabilized multilevel Monte Carlo method and the standard Monte Carlo method using S-ROCK as numerical integrator. The space stepsize is set to $\Delta x=1/40$ and the finest time stepsize varies between 2^{-7} and 2^{-20}. It can be observed that for small time stepsizes h_L the stabilized MLMC method reduces the computational cost significantly compared to the standard MC method.

In Fig. 9 the mean of $u(t,x)$ is approximated at $t=1$ using stabilized multilevel Monte Carlo. The finest time stepsize is chosen as $h_L=2^{-10}$. For the space discretization $\Delta x=1/40$ (see Fig. 9)) and $\Delta x=1/80$ (see Fig. 9(b)) are used, respectively. In addition, on each plot a single trajectory of $u(t,x)$ at $t=1$ using S-ROCK and $h=2^{-10}$ is added. Note that an approximation of the mean of $u(t,x)$ using the standard Monte Carlo method with Euler–Maruyama would require in the case $\Delta x=1/40$ a time stepsize smaller than 3.1×10^{-4} and a computational cost of approximately 2.6×10^{12} function evaluations. In the case $\Delta x=1/80$, the time stepsize would have to be smaller than 7.8×10^{-5} and the corresponding computational cost would be about 3.4×10^{14} function evaluations.

Fig. 10 illustrates an approximation of the second moment of $u(t,x)$ at $t=1$ using the stabilized multilevel Monte Carlo method with finest time stepsize $h_L=2^{-10}$ and space discretization $\Delta x=1/40$ and $\Delta x=1/80$, respectively. Furthermore, a dotted line represents a single trajectory of the approximation of $u(1,x_i)^2$ using S-ROCK with $h=2^{-10}$.

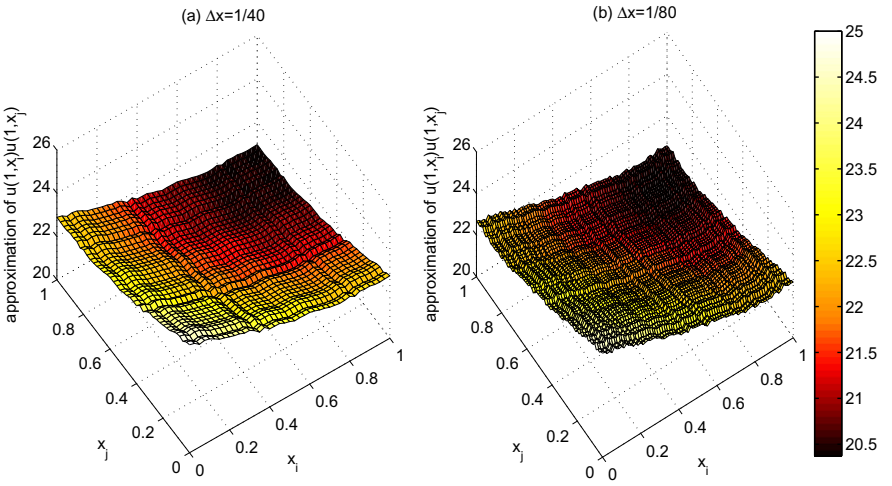

Fig. 12. Stochastic partial differential equation: Single trajectory of $u(t,x_i)u(t,x_j)$ at $t=1$ using S-ROCK with $h=2^{-10}$, $\Delta x=1/40$ (see (a)) and $\Delta x=1/80$ (see (b)).

In Fig. 11 approximations of $\mathbb{E}\big[u(t,x_i)u(t,x_j)\big]$ at $t=1$ with $i,j\in\{0,1,\dots,1/\Delta x\}$ are shown. As approximation procedure the stabilized MLMC method with $L=10$ and space discretization $\Delta x=1/40$ and $\Delta x=1/80$, respectively, is used. Single trajectories of S-ROCK approximations of $u(t,x_i)u(t,x_j)$ at $t=1$ are illustrated in Fig. 12.

5. Conclusion

We have presented a new stabilized multilevel Monte Carlo method for mean square stable SDEs with multiple scales. We have shown that the standard MLMC method fails to achieve the optimal computational complexity $\mathcal{O}(\varepsilon^{-2}(\log(\varepsilon))^2)$ to compute the expectation of functionals with an accuracy of $\mathcal{O}(\varepsilon)$ as some or all the sequence of stepsizes needed in the MLMC method are not accessible due to stepsize restriction. In the worst case, only a standard Monte Carlo method can be used and the computational complexity can deteriorate to $\mathcal{O}(\varepsilon_{\mathrm{MC}}^{-3})$, where $\varepsilon_{\mathrm{MC}}$ is smaller than ε, the desired accuracy. We have then shown that using the S-ROCK methods, a family of stabilized methods based on the Euler–Maruyama scheme, it is possible to define a stabilized MLMC method that is applicable for stiff mean square stable problems. By an optimal choice of the stabilization procedure, varying from the coarse to the fine MLMC levels, we showed that it is possible to recover the optimal complexity of the MLMC for nonstiff problems up to a factor involving the square root of a quantity called the stiffness parameter. Even though our stability analysis rely on the usual linear scalar SDE used to characterize mean square-stability of numerical integrators, we have shown through numerical experiments on multidimensional nonlinear noncommutative stiff SDEs and on a system of SDEs obtained from a space-discretized SPDE that our new stabilized MLMC method is efficient also for more general problems.

Acknowledgements

The research of A.A. and A.B. is partially supported by Swiss National Foundation Grant 200021_140692.

References

[1] P. Glasserman, Monte Carlo methods in financial engineering, volume 53 of Applications of Mathematics (New York), Springer-Verlag, New York, 2004.
[2] S. Heinrich, Monte Carlo complexity of global solution of integral equations, J. Complex. 14 (1998) 151–175.
[3] M. Giles, Multilevel Monte Carlo path simulation, Oper. Res. 56 (3) (2008) 607–617.
[4] G. Maruyama, Continuous markov processes and stochastic equations, Rend. Circ. Mat. Palermo 4 (1955) 48–90.
[5] D. Higham, An algorithmic introduction to numerical simulation of stochastic differential equations, SIAM Rev. 43 (3) (2001) 525–546.
[6] Y. Saito, T. Mitsui, Stability analysis of numerical schemes for stochastic differential equations, SIAM J. Numer. Anal. 33 (1996) 2254–2267.
[7] Y. Cao, D. Gillespie, L. Petzold, Adaptive explicit-implicit tau-leaping method with automatic tau selection, J. Chem. Phys. 126 (224101) (2007) 1–9.
[8] P. Rué, J. Villà-Freixa, K. Burrage, Simulation methods with extended stability for stiff biochemical Kinetics, BMC Syst. Biol. 4 (110) (2010) 1–13.
[9] K.D. Elworthy, A. Truman, H.Z. Zhao, J.G. Gaines, Approximate travelling waves for generalized KPP equations and classical mechanics, Proc. Roy. Soc. London Ser. A 446 (1928) (1994) 529–554.
[10] E. Platen, N. Bruti-Liberati, Numerical solution of stochastic differential equations with jumps in Finance, volume 64 of Stoch, Model. and Appl. Prob. Springer, Berlin Heidelberg, 2010.
[11] A. Abdulle, W. E, T. Li, Effectiveness of implicit methods for stiff stochastic differential equations, Commun. Comput. Phys. 3 (2) (2008) 295–307.
[12] E. Vanden-Eijnden, Numerical techniques for multiscale dynamical system with stochastic effects, Commun. Math. Sci. 1 (2003) 385–391.
[13] A. Abdulle, A. Barth, C. Schwab. Multi–Level Monte Carlo heterogenenous multiscale finite element method for elliptic pdes with stochastic multiscale coefficients, SIAM Multiscale Model. Simul., in press.
[14] A. Abdulle, On roots and error constants of optimal stability polynomials, BIT Numer. Math. 40 (1) (2000) 177–182.

[15] A. Abdulle, A. Medovikov, Second order Chebyshev methods based on orthogonal polynomials, Numer. Math. 90 (1) (2001) 1–18.
[16] A. Abdulle, Fourth order Chebyshev methods with recurrence relation, SIAM J. Sci. Comput. 23 (6) (2002) 2041–2054.
[17] T. Dumont, M. Duarte, S. Descombes, M.-A. Dronne, M. Massot, V. Louvet, Simulation of human ischemic stroke in realistic 3D geometry, Commun. Nonlinear. Sci. Numer. Simulat. 18 (6) (2013) 1539–1557.
[18] A. Abdulle, G. Vilmart, PIROCK: a swiss-knife partitioned implicit-explicit orthogonal Runge–Kutta Chebyshev integrator for stiff diffusion-advection-reaction problems with or without noise, J. Comput. Phys. 242 (2013) 869–888.
[19] A. Ern, M. Vohralik, Adaptive inexact newton methods with a posteriori stopping criteria for nonlinear diffusion pde, SIAM J. Sci. Comput., in press, 2013.
[20] A. Abdulle, S. Cirilli, S-ROCK: Chebyshev methods for stiff stochastic differential equations, SIAM J. Sci. Comput. 30 (2) (2008) 997–1014.
[21] A. Abdulle, T. Li, S-ROCK methods for stiff Ito SDEs, Commun. Math. Sci. 6 (4) (2008) 845–868.
[22] A. Abdulle, G. Vilmart, K. Zygalakis. Weak second order explicit stabilized methods for stiff stochastic differential equations, SIAM SISC, in press, 2013.
[23] M. Hutzenthaler, A. Jentzen, P. Kloeden, Divergence of the multilevel Monte Carlo method. ArXiv, preprint, 1105.0226, 2011.
[24] G. Milstein, M. Tretyakov, Stochastic numerics for mathematical physics, Scientific Computing, Springer-Verlag, Berlin and New York, 2004.
[25] P. Kloeden, E. Platen, Numerical Solution of Stochastic Differential Equations, Springer-Verlag, Berlin and New York, 1992.
[26] L. Arnold, Stochastic Differential Equations: Theory and Applications, John Wiley and Sons, New York, 1974.
[27] A. Abdulle, S. Cirilli, Stabilized methods for stiff stochastic systems, C. R. Math. Acad. Sci. Paris, 345 (10) (2007) 593–598.
[28] P. Van der Houwen, B. Sommeijer, On the internal stage Runge–Kutta methods for large m-values, Z. Angew. Math. Mech. 60 (1980) 479–485.
[29] D. Lamberton, B. Lapeyre, Introduction to Stochastic Calculus Applied to Finance, Ellipses, Paris, 1997.
[30] Y. Saito, T. Mitsui, Mean-square stability of numerical schemes for stochastic differential systems, Vietnam J. Math. 30 (Suppl.) (2002) 551–560.
[31] A. Rathinasamy, K. Balachandran, Mean-square stability of second-order Runge–Kutta methods for multi-dimensional linear stochastic differential systems, J. Comput. Appl. Math. 219 (1) (2008) 170–197.
[32] E. Buckwar, C. Kelly, Towards a systematic linear stability analysis of numerical methods for systems of stochastic differential equations, SIAM J. Numer. Anal. 48 (1) (2010) 298–321.
[33] X. Mao, Stochastic stabilization and destabilization, Syst. Contr. Lett. 23 (4) (1994) 279–290.
[34] D.J. Higham, X. Mao, Nonnormality and stochastic differential equations, BIT 46 (3) (2006) 525–532.
[35] E. Hairer, G. Wanner, Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems, Springer-Verlag, Berlin and Heidelberg, 1996.
[36] J. Murray, Mathematical Biology I: An Introduction, Springer, Seattle, 2002.
[37] A. Barth, A. Lang, Multilevel Monte Carlo method with applications to stochastic partial differential equations, Int. J. Comput. Math. 89 (18) (2012) 2479–2498.
[38] M. Giles, C. Reisinger, Stochastic finite differences and multilevel Monte Carlo for a class of SPDEs in finance, SIAM J. Finan. Math. 3 (1) (2012) 572–592.