Stepsize control for the Milstein scheme using first-exit-times

H Lamba
Department of Mathematical Sciences,
George Mason University, MS 3F2,
4400 University Drive,
Fairfax, VA 22030 USA

Abstract

We introduce a variable timestepping algorithm for the Milstein scheme applied to SDEs with a scalar stochastic forcing. Multiple local error estimates are used, corresponding to different terms appearing in the Taylor series of the local truncation error. The timesteps are then chosen so as to bound the standard deviation of the contribution of each of these terms over a unit time interval, analogous to the Error-Per-Unit-Step control used for ODE solvers. The problem of determining the optimal timestep then translates to one of finding the first-exit distribution of the Wiener process from a simple boundary and eliminates the need for timestep rejections.

1 Introduction

We shall consider the numerical approximation of strong (pathwise) solutions of stochastic differential equations (SDEs) forced by a scalar Wiener process written in Itô form as

$$dX = f(X)\,dt + g(X)\,dW \qquad X(0) = X_0 \tag{1}$$

where $X \in \mathbb{R}^m$. The strong solutions exist and are unique under various assumptions on the drift and diffusion coefficients. In addition, we shall only require minimal smoothness conditions, namely that $f, g \in C^2(\mathbb{R}^m)$.

SDEs have rapidly become important modeling tools in many scientific disciplines as the significance of stochastic effects has become more widely recognized and the computational and theoretical tools to deal with them have become more sophisticated. However there is still a great deal of work remaining to develop efficient and reliable SDE software. Indeed there is still a lack of consensus even on basic strategy with several, fundamentally different, approaches to the problem appearing in the literature in the last decade or so.

Once an SDE scheme has been chosen, the efficiency of the scheme can be improved by allowing the timesteps used to change adaptively as the path progresses. Two early papers by Gaines and Lyons [5] and Mauthner [21] stimulated much of the work on adaptivity that followed. They built on previous theoretical results to provide algorithms whose convergence was guaranteed under certain conditions even allowing for the possibility of step-size rejections and non-stopping times. Both of these papers used very simple halving-and-doubling strategies for the timestep selection but, even so, demonstrated the computational advantages of adaptivity. In an interesting sequence of papers, Hofmann, Müller-Gronbach and Ritter [9, 10, 11] used a choice of timestep inversely proportional to $\|g(X)\|$ and showed that this is optimal in a precise sense related to the number of samples of the Wiener process that are required. Note that this strategy only controls upon the diffusion. In [19] the opposite approach was used and the timestep was controlled using a local error estimate based only upon the drift. In [2, 3, 4], the approach taken was to develop and employ embedded stochastic Runge-Kutta schemes for SDEs and then choose timesteps based upon asymptotic formulae designed for adaptivity in ODEs. Other related work includes adaptivity designed for weak-noise systems [24] such as occur in integrated circuit manufacturing and adaptivity for weak-solution algorithms [29].

The approach taken here extends an idea introduced in [17] where two local error estimates for the Milstein scheme were limited by a user-defined tolerance τ. One estimate, loosely speaking, monitored the deterministic integration and the other the stochastic one. This enabled the algorithm to behave differently in different regimes, for example the cases of strong or weak diffusion. In addition, under further reasonable assumptions on the error control, desirable mean-square stability properties could be proved for a class of linear multiplicative-noise test problems [20].

Three different modifications will be made here to the algorithm proposed in [17]. Firstly, in Section 2 an additional stochastic error control is incorporated. Secondly, in Section 3 an alternative error control strategy is introduced for the (two) stochastic error estimates. This is a stochastic counterpart to the Error-Per-Unit-Step (EPUS) criterion for ODEs and requires that the *standard deviation* of the error estimates over a unit time interval is bounded by τ (the mean is 0). Finally, in Section 4 a novel timestep selection strategy is employed. This relies on the fact that the optimal timestep, subject to satisfying the error controls, can be defined as the first-exit-time of the Wiener process from a simple rectangular boundary. The probability distribution over this boundary is complicated but can be effectively reduced to a small number of look-up tables used in conjunction with a uniformly-distributed random variable (note that no Gaussian variables need to be generated and no stepsize rejections can occur). Section 5 contains some numerical examples.

2 Error Estimates

In order to simplify the analysis slightly, we start by rewriting (1) in Stratonovich form as

$$dX = \underline{f}(X)\,dt + g(X) \circ dW \tag{2}$$

where $\underline{f} = f - \frac{1}{2}g'g$ is the Stratonovich drift. The Milstein scheme, using a timestep h and corresponding Wiener increment ΔW, is defined by

$$X_{n+1} = X_n + \Delta W g(X_n) + h\underline{f}(X_n) + \frac{1}{2}|\Delta W|^2 g'(X_n)g(X_n) \tag{3}$$

and advances the solution from X_n at time t_n to X_{n+1} at time $t_{n+1} = t_n + h$. It has strong-order 1 and the truncation error can be expanded as

$$\underbrace{J_{10}\underline{f}'g + J_{01}g'\underline{f} + \frac{1}{6}(\Delta W)^3 g''g^2 + \frac{1}{6}(\Delta W)^3 g'^2 g}_{\mathcal{O}(h^{\frac{3}{2}})} \quad + \quad \frac{1}{2}h^2\underline{f}'\underline{f} + \ldots \tag{4}$$

where the Stratonovich integrals J_{01} and J_{10} are defined by

$$\begin{aligned} J_{01} &= \int_{t_n}^{t_{n+1}} \int_{t_n}^{s_1} ds_1 \circ dW, \\ J_{10} &= \int_{t_n}^{t_{n+1}} \int_{t_n}^{s_1} \circ dW \; ds_1. \end{aligned} \tag{5}$$

For our error estimates we shall select certain of the terms appearing in the truncation error (4).

At this point it is worth commenting upon a fundamental difference between (4) and the corresponding truncation error expansions for ODE schemes which have the form

$$\frac{h^{p+1}}{(p+1)!}(\text{lin. comb. elementary differentials}) + \mathcal{O}(h^{p+2}) \tag{6}$$

for an order-p method. In order to achieve successful adaptive timestepping for ODEs it is not necessary to know which of the leading-order terms are dominating the truncation error. This is because all the leading-order elementary differentials are multiplied by the same timestep-dependent quantity h^{p+1}. Suppose that some local error estimate $E(X, h)$ is being controlled by enforcing

$$E(X_n, h) \leq \tau h^\rho \tag{7}$$

i.e. a timestep h is only accepted if (7) is satisfied. The choice of $\rho = 0$ corresponds to Error-Per-Step (EPS) control while $\rho = 1$ is Error-Per-Unit-Step (EPUS) control. A new candidate timestep h' (for the same timestep if the control (7) was violated) can be found by using variations upon the asymptotic formula

$$h' = \theta h \left(\frac{\tau}{E(X_n, h)} \right)^{\frac{1}{p-\rho}} \tag{8}$$

where θ is a 'safety factor' slightly less than 1. Such variations include maximum-allowable timesteps, maximum ratios between adjacent timesteps (see [7, 26]) and ideas from control-theory [6] that result in highly-efficient and robust algorithms that are easy to implement with low computational overheads. Also, as $\tau \to 0$ the number of stepsize rejections also goes to zero and the error estimate is approximately a factor of θ within its optimal value. Typically the error estimate $E(X_n, h)$ is derived as the difference between two

approximate solutions. An illustrative example is that of embedded Runge-Kutta pairs of orders $p-1$ and p where the difference is an estimate for the error committed by the lower-order method. However, possible stability considerations aside, it makes sense to advance the solution using the already computed higher-order method — this is known as *extrapolation*. The error estimate is now only loosely-related to the true truncation error but this is enough to guarantee convergence (apart from some exceptional special cases [27, 16]) and retain the efficiency advantages of adaptivity which are very impressive and almost independent of the order p.

We now return to the Milstein truncation error (4). Different elementary differentials at $\mathcal{O}(h^{\frac{3}{2}})$ are now multiplied by different timestep-dependent quantities, namely J_{10}, J_{01} and ΔW^3, which all scale differently with the choice of h. This suggests that in order to efficiently control the one-step error it is now desirable to estimate which of the terms dominate(s) the local error (in contrast to the ODE adaptive strategy) and choose candidate timesteps accordingly.

In principle, all four of the leading-order error terms can be computed or estimated but we shall restrict ourselves to incorporating just two of them, namely

$$E_s(X_n, h) := \frac{1}{6}\|\Delta W^3 g'^2 g\|_\infty \tag{9}$$

and

$$E_t(X_n, h) := \|J_{10}\underline{f}' g \Delta W\|_\infty \tag{10}$$

where the choice of ∞-norm helps reduce the computational cost when $m > 1$ (even simpler upper bounds for these quantities could be used instead, such as replacing $\|\underline{f}' g\|_\infty$ with $\|\underline{f}'\|_\infty \|g\|_\infty$). We shall refer to E_s and E_t as the stochastic error estimates. Note that the computation of E_t requires that $\underline{f}'$ be computed exactly or estimated via a finite difference approximation. All the other function evaluations are needed for the calculation of the Milstein approximation itself and so are already available.

We now introduce one further error estimate based only upon the drift coefficient. The motivation for this is that in the weak-diffusion limit, for typical user-defined tolerances, the leading-order error may in fact be the term $\frac{1}{2}h^2\underline{f}'\underline{f}$ appearing at $\mathcal{O}(h^2)$ in the expansion (4). This leads to the deterministic error estimate

$$E_d := \frac{1}{2}h^2\|\underline{f}'\underline{f}\|_\infty. \tag{11}$$

In [17], only E_s and E_d were used to generate adaptive strategies. The resulting algorithms were flexible enough to operate differently depending upon whether the drift or diffusion error was dominating. In the former case the timestep selection strategy was close to that used for ODEs while in the latter case a more complicated prototype selection procedure was developed. In [20] an error control strategy similar to EPUS was used on the class of linear test problems with multiplicative noise. The resulting algorithm was shown to have very desirable mean-square stability properties (in fact, the multiplicative noise analog of A-stability for ODEs). The reader is directed towards [20] for further details.

As mentioned in Section 1 the work detailed here extends and improves the ideas of [17] in three significant ways. The first of these was the inclusion of the error control E_t. This ensures that the true one-step truncation error is accurately monitored for a wider class of SDEs. In particular, SDEs with additive noise were not satisfactorily covered by the algorithm of [17] because $E_s \equiv 0$. This meant that although the algorithm would still converge to the correct pathwise solution as $\tau \to 0$ it would do so by controlling only the deterministic error control E_d which is certainly non-optimal if the diffusion strength is significant. In the next section the use of an EPUS error control over single timesteps for the stochastic estimates E_s and E_t is replaced by controlling the root-mean-square error over a unit time-interval.

3 Error-Per-Unit-Time Controls

We now consider each of the three error estimates in isolation and derive controls to bound the estimated cumulative errors over a unit time interval. The implementation of EPUS control for the deterministic error estimate E_d is fundamentally the same as for ODEs. The functions $\underline{f}, \underline{f}'$ are assumed to be constant, an as-yet-undetermined timestep h is to used on each step, and for this choice of h the cumulative error over a unit time interval should be bounded by τ giving

$$\sum_{1}^{N_d} \frac{1}{2} h^2 \|\underline{f}' \underline{f}\|_\infty \leq \tau$$

where $N_d = 1/h$. This simplifies to $h \leq \frac{2\tau}{\|\underline{f}' \underline{f}\|_\infty}$ and thus provides an upper bound for the timestep h with respect to the deterministic error control. In

practice relative, absolute or mixed error controls may be used and we allow for this degree of generality by replacing the tolerance τ with functions $\sigma_d(X,\tau), \sigma_s(X,\tau)$ and $\sigma_t(X,\tau)$ for the three different controls. So the deterministic error control leads to the following upper bound on the timestep

$$h \leq h_d := \frac{2\sigma_d(X_n,\tau)}{\|\underline{f}'(X_n)\underline{f}(X_n)\|_\infty}. \tag{12}$$

Note that because the estimate E_d was defined as an explicit function of h that has been inverted, an explicit bound for h has now been found and stepsize rejections do not occur.

We now turn our attention to the two stochastic error controls starting with E_s. By analogy with the above argument, assuming that g and g' are constant, it is the cumulative error due to the term $\frac{1}{6}\Delta W^3 g'^2 g$ over a unit time interval that should be controlled. This is a random variable with mean zero and so we control the standard deviation at time $t+1$ by enforcing

$$\mathrm{SD}\left(\sum_{i=1}^{N_s}\frac{1}{6}\Delta W_i^3\|g'^2 g\|_\infty\right) \leq \sigma_s(X_n,\tau) \tag{13}$$

where the summation is over all the N_s timesteps taken during the unit time interval.

However the quantity E_s depends upon the Wiener increments ΔW_i, not the timestep length h and the optimal choice of timesteps should reflect this. To this end let us introduce a constant $C_s > 0$ and hypothesize an error control of the form

$$|\Delta W_i| \leq C_s \qquad \forall i = 1,\ldots,N_s. \tag{14}$$

A 'perfect' timestep selection procedure is one that, at negligible cost, identifies the sequence of timesteps h_i such that $\forall i = 1,\ldots,N_s, h_i$ is the smallest timestep where equality occurs in (14). Thus the timestep h_n used to advance from the integration time t_n is the first-exit-time of the Wiener Process from the interval $[W(t_n)-C_s, W(t_n)+C_s]$. The probability distribution of this exit time is well-known [14] and has mean $\mathbf{E}(h_n) = C_s^2$.

Using such an error control the quantity on the left hand side of (13) is the sum of N_s random variables taking the two possible values $\pm\frac{1}{6}C_s^3\|g'^2 g\|_\infty$

each with probability $\frac{1}{2}$. If we ignore the distribution of N_s and assume that

$$N_s = 1/\mathbf{E}(h_i) = \frac{1}{C_s^2}$$

then the sum in (13) is a random walk with standard deviation

$$\frac{1}{6}C_s^3\|g'^2 g\|_\infty\sqrt{N_s} = \frac{1}{6}C_s^2\|g'^2 g\|_\infty.$$

Finally, forcing equality in (13) and introducing a function $\sigma_s(X_n, \tau)$ leads to

$$C_s := \sqrt{\frac{6\sigma_s(X_n,\tau)}{\|g'^2 g\|_\infty}}$$

and our second error control

$$|\Delta W_n| \le C_s = \sqrt{\frac{6\sigma_s(X_n,\tau)}{\|g'^2 g\|_\infty}}. \tag{15}$$

We shall refer to this type of control as Error-Per-Unit-Time (EPUT) to distinguish it from EPUS.

It only remains to find the corresponding EPUT error control for the quantity E_t. Our starting point is the following characterization of J_{10} over an interval of length h [15],

$$J_{10} = \frac{1}{2}h\left(\Delta W + \frac{\eta}{\sqrt{3}}\right) \tag{16}$$

where $\eta \sim \mathcal{N}(0, h)$ is independent of ΔW. An EPUT error control corresponds to enforcing

$$\begin{aligned}
&\mathrm{SD}\left(\sum_{i=1}^{N_t}\frac{1}{2}h_i\|\underline{f}'g\|_\infty\left(\Delta W_i + \frac{\eta_i}{\sqrt{3}}\right)\right) \le \sigma_t(X_n,\tau)\\
\Rightarrow\ &\mathrm{Var}\left(\sum_{i=1}^{N_t}\frac{1}{2}h_i\left(\Delta W_i + \frac{\eta_i}{\sqrt{3}}\right)\right) \le \frac{\sigma_t(X_n,\tau)^2}{\|\underline{f}'g\|_\infty^2}\\
\Rightarrow\ &\mathrm{Var}\left(\sum_{i=1}^{N_t}\frac{1}{2}h_i\Delta W_i\right) + \mathrm{Var}\left(\sum_{i=1}^{N_t}\frac{1}{2}h_i\frac{\eta_i}{\sqrt{3}}\right) \le \frac{\sigma_t(X_n,\tau)^2}{\|\underline{f}'g\|_\infty^2}
\end{aligned} \tag{17}$$

As with E_s, we shall assume a perfect control of the form $|\Delta W| = C_t$ for some constant $C_t > 0$. Now, for each $i = 1, \ldots, N_t$, the h_i and η_i are independent and $\mathbf{E}(h_i^2) = \frac{5}{3}C_t^4$ giving

$$\begin{aligned}\mathrm{Var}(h_i\Delta W_i) &= \mathbf{E}(h_i^2\Delta W_i^2) - \mathbf{E}(h_i\Delta W_i)^2 \\ &= \mathbf{E}(h_i^2\Delta W_i^2) \\ &= C_t^2\mathbf{E}(h_i^2) \\ &= \frac{5}{3}C_t^6.\end{aligned}$$

Then assuming $N_t = 1/C_t^2$ and summing allows us to replace the first term in (17) with $\frac{5}{12}C_t^4$.

To calculate the second term of (17), we use the independence of h_i, η_i and $\mathbf{E}(h_i^3) = \frac{61}{15}C_t^6$ to obtain

$$\begin{aligned}\mathrm{Var}(h_i\eta_i) = \mathbf{E}(h_i^2\eta_i^2) &= \int_{x=0}^{\infty} \mathbf{E}(h^2\eta^2|h = x)\mathrm{Pr}(h = x)\ dx \\ &= \int_{x=0}^{\infty} \mathbf{E}(x^2\eta^2|h = x)\mathrm{Pr}(h = x)\ dx \\ &= \int_{x=0}^{\infty} x^2\mathbf{E}(\eta^2|h = x)\mathrm{Pr}(h = x)\ dx \\ &= \int_{x=0}^{\infty} x^3\mathrm{Pr}(h = x)\ dx \\ &= \mathbf{E}(h^3) = \frac{61}{15}C_t^6\end{aligned}$$

Summing over the N_t steps gives the second variance in (17) as $\frac{61}{180}C_t^4$. Combining these results in (17) and enforcing equality finally results in an optimal value of $C_t = \sqrt{\frac{45\sigma_t(X_n,\tau)}{34\|\underline{f}'g\|_\infty}}$ which we approximate as

$$C_t := 1.15\sqrt{\frac{\sigma_t(X_n,\tau)}{\|\underline{f}'g\|_\infty}}$$

giving our final error control

$$|\Delta W_n| \leq C_t = 1.15\sqrt{\frac{\sigma_t(X_n,\tau)}{\|\underline{f}'g\|_\infty}}. \tag{18}$$

In order to satisfy both (15) and (18) we define

$$C = \min(C_s, C_t). \tag{19}$$

Our optimal choice of timestep h_n to advance from a time t_n whilst obeying all the error controls is now given by

$$h_n = \sup_{h>0} \left(h : h \leq h_d \text{ and } |W(t_n + t) - W(t_n)| \leq C \; \forall 0 < t < h \right). \tag{20}$$

Many of the previous approaches to adaptivity have either attempted to directly apply ideas from the ODE setting or have focused upon the drift or the diffusion integral. The philosophy motivating this new approach is that on each timestep the Brownian increment ΔW plays a similar role for the stochastic integral in (2) that the steplength h plays for the deterministic one. Then, from first principles, these quantities are treated in a consistent manner to arrive at (20).

To conclude this section we note that the two neglected terms in (4) can be analyzed in a similar fashion resulting in the monitoring of all the leading-order truncation terms. This leads to two additional upper bounds on $|\Delta W|$ and the smallest of all four of them would be used to define C and control the timestep. More extensive numerical testing may conclude that incorporating these extra controls justifies the computational cost.

4 Optimal Timesteps as First-Exit-Times

The definition of the optimal timestep (20) can be interpreted as the first-exit-time of the Brownian path from the boundary shown in Figure 1. The path leaving the upper or lower boundaries (L and U) corresponds to one of the stochastic error controls limiting the timestep while exiting the vertical boundary V at $t = t_n + h_d$ occurs when the deterministic error control dominates. The analysis of the previous section assumes that the optimal timesteps can be identified and we now show that this is indeed possible in a computationally efficient way.

The first step is to use the scaling property of Brownian motion which states that the transformed process

$$W'_{t'} = \frac{1}{\sqrt{a}} W_{t'/a}$$

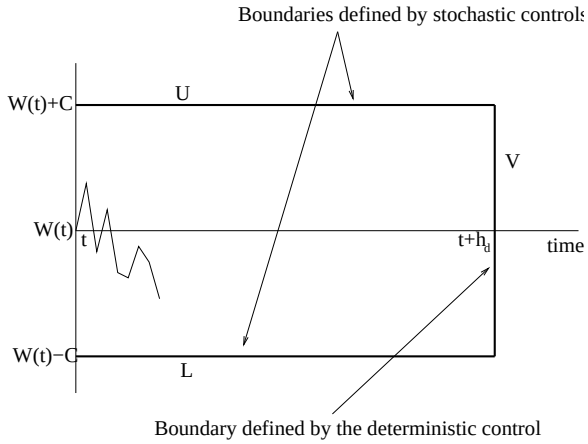

Figure 1: The timestep used will be the first-exit-time from the rectangle shown.

is another Brownian motion for any $a > 0$. Choosing $a = C^2$ results in a rescaled exit-time problem for a standard Brownian motion with upper and lower boundaries, L' and U', at $W'_t = \pm 1$ and vertical boundary V' at $h'_d = h_d/C^2$.

The probability and cumulative density functions for the first exit points over the transformed boundary have a known analytical form [1, 8]. The cumulative probability of first exiting via either L' or U' before time t' is given by

$$P(t', L' \text{ or } U') = 2\sum_{j=0}^{\infty}(-1)^j \,\overline{\Phi}\left(\frac{2j+1}{\sqrt{t'}}\right), \qquad \forall t' > 0 \tag{21}$$

where $\overline{\Phi}$ is the complement of the standard error function. The CDF is plotted in Figure 2.

The probability of exiting V' with $W_{h'_d} \in [-1, x']$ is

$$\begin{aligned} P(x') \;=\; & \frac{1}{\sqrt{h'_d}}\int_{-1}^{x'}\left[\phi\left(\frac{y}{\sqrt{h'_d}}\right) + \sum_{j=1}^{\infty}\left\{\phi\left(\frac{y-4j}{\sqrt{h'_d}}\right) - \phi\left(\frac{y+4j-2}{\sqrt{h'_d}}\right)\right.\right. \\ & \left.\left. +\phi\left(\frac{y+4j}{\sqrt{h'_d}}\right) - \phi\left(\frac{y-4j+2}{\sqrt{h'_d}}\right)\right\}\right] dy, \qquad x' \in [-1, 1]. \end{aligned} \tag{22}$$

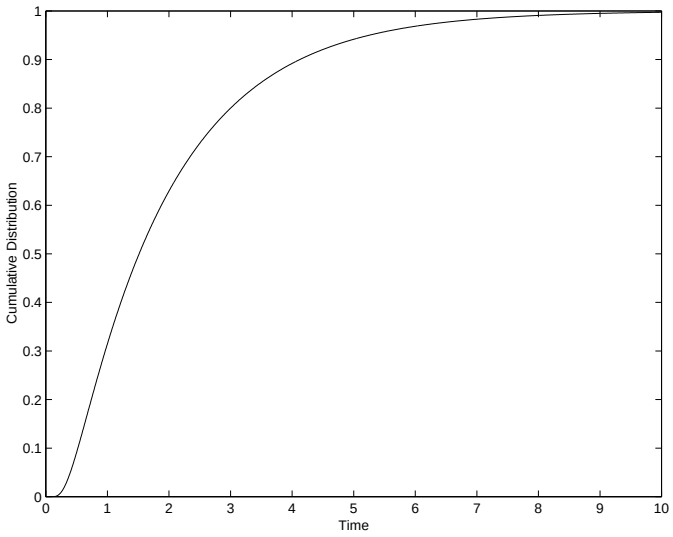

Figure 2: The cumulative distribution along the upper and lower boundaries taken together for $C = 1$.

The inverses of these cumulative density functions can now be used to generate first-exit points. This is achieved by providing look-up tables although approximating functions can in principle be used if they are sufficiently cheap to evaluate to the required accuracy. In order to reduce the number of look-up tables required by the algorithm, only a small number of values of h'_d are permitted and so h'_d is reduced to the nearest allowable value.

A random variable $z \sim U[0,1]$ is generated. If $z \leq P(h'_d, L' \text{ or } U')$ then a look-up table for the inverse of the cdf (21) is used to determine the corresponding exit-time and an extra random bit chooses whether $\Delta W' = +1$ or -1. If $z > P(h'_d, L' \text{ or } U')$ then $z - P(h'_d, L' \text{ or } U')$ is used in the appropriate look-up table for the inverse of (22) which fixes the value of $\Delta W' \in [-1, 1]$. Then the scaling transformation is inverted to obtain the values of h and ΔW required for updating the algorithm.

An examination of Figure 2 suggested the quantized values

$$h'_d = 0.1, 0.2, \ldots, 0.9, 1, 2, 3, 4, 5.$$

No claim is made for the optimality of these values but they appear to be sufficient for initial testing purposes. A maximal value of 5 prevents the

algorithm taking abnormally large timesteps caused by unusually 'flat' segments of Brownian motion. On the other hand if the transformed value of $h_d < 0.1$ this corresponds to a weak-diffusion process and the stochastic error controls are simply ignored. That is, the next timestep h is chosen to be h_d irrespective of the value of ΔW which is now chosen from $\mathcal{N}(0, h_d)$ in the usual way. On any given step there is a very low probability, less than $P(0.1, L' \text{ or } U') \approx 0.000016$, of exceeding the stochastic error bound and this is deemed to be acceptably small.

This completes the description of the algorithm. Apart from the case of very weak diffusion described above, no Gaussian variables need to be calculated — just a uniform random variable and a random bit. The only extra function evaluations needed over and above those required by the Milstein method itself are the evaluations or estimations of $\underline{f}'$, at most one per step. Also, the lack of stepsize rejections means that intermediate values of ΔW conditioned upon evaluations at later times are not required. Thus the computational overheads required to implement the algorithm are quite small.

Finally, we point out that the use of first-exit-times described here may have additional advantages for Monte-Carlo simulations of SDEs involving barriers. These are problems where the first hitting time of the solution X_t with a known boundary is either required to advance the solution or is needed to calculate some function of the solution path. When the solution is close to such a barrier the values of C and h_d can be reduced to ensure that the barrier is not breached during the timestep (algorithms for such problems using exponentially-distributed random timesteps have been proposed in [12, 13]).

5 Numerical Results

We now present numerical results on three test problems using the proposed algorithm. No attempt has been made to optimize the performance of the algorithm with respect to hardware or software-specific details. Also, we do not compare the computational cost of each timestep against other algorithms since these are also likely to be implementation specific and in addition will depend upon the complexity and nature of the SDE. Instead we measure the number of timesteps used for various values of the tolerance τ and compare

Table 1: Numerical results for Test Problem 1.

τ	Method	Av. # Steps used	Error
10^{-1}	Variable	32	0.17
	Fixed		0.26
10^{-2}	Variable	307	0.014
	Fixed		0.039
10^{-3}	Variable	2810	0.0017
	Fixed		0.0030

the error committed with that of the Milstein method operating in fixed timestep mode using the same number of steps.

The class of scalar, multiplicative noise test problems defined, in Itô form, by

$$dX = \lambda X\ dt + \mu X dW, \qquad X(0) = X_0 \neq 0 \tag{23}$$

will provide our first numerical example. The exact solution is

$$X(t) = X_0 \exp\left((\lambda - \frac{1}{2}\mu^2)t + \mu W(t)\right)$$

and mean-square stability [25] occurs when $\Re(\lambda)+\frac{1}{2}|\mu^2| < 0$. The Stratonovich drift is $\underline{f}(x) = (\lambda - \frac{1}{2}\mu^2)x$. The parameters used are $\lambda = -1, \mu = 2, X(0) = 1$ and the scheme employs absolute error controls $\sigma_s = \sigma_d = \sigma_t = \tau$ over the time interval $[0, 1]$.

The results are shown in Table 1 and were obtained by averaging over 100 runs. The error value recorded is the average maximum absolute error committed over each run. The average number of timesteps used is also shown. Then the fixed timestepping Milstein method with the same steplength as the average used by the adaptive scheme was run 100 times and the average maximum absolute error was calculated.

The use of the adaptive scheme reduces the average maximum error by approximately a factor of 2. However there are also stability considerations and, as for ODEs, adaptivity can impart desirable stability properties [28, 18]. In fact, a straightforward modification of the argument presented in [20][Section 4] shows that, using relative error controls, the above algorithm preserves the mean-square stability properties of (23).

Table 2: Numerical results for Test Problem 2.

τ	Method	Av. # Steps used	Error
10^{-2}	Variable	5	0.012
	Fixed		0.026
10^{-3}	Variable	42	0.0012
	Fixed		0.0034
10^{-4}	Variable	410	0.00011
	Fixed		0.00042

Table 3: Numerical results for Test Problem 2 using the Itô drift.

τ	Method	Av. # Steps used	Error
10^{-2}	Variable	19	0.0021
	Fixed		0.067
10^{-3}	Variable	190	0.00023
	Fixed		0.00089
10^{-4}	Variable	1800	0.000019
	Fixed		0.00009

The second test problem is

$$dX = \frac{1}{3}X^{\frac{1}{3}}\,dt + X^{\frac{2}{3}}dW, \qquad X(0) = 1 \tag{24}$$

and has exact solution $X(t) = (1 + \frac{1}{3}W(t))^3$. Note that for this problem the Stratonovich drift $\underline{f}(x) \equiv 0$ and so the algorithm just uses the two stochastic controls. As before, absolute error controls are used. The results are presented in Table 2. Note that there is approximate tolerance proportionality in both the error committed and the number of timesteps used as τ is decreased.

In general the Stratonovich drift will not vanish and sometimes the Itô drift will be either more readily available and/or cheaper to compute. It was therefore of interest to modify the algorithm to use the function f rather than $\underline{f}$ in the deterministic control (12). Table 3 shows that, for a given tolerance, smaller timesteps are taken but there is no significant improvement in the overall efficiency on a per-step basis.

This test problem was also solved using the same three error controls but

Table 4: Numerical results for Test Problem 2 using EPS error control.

τ	Method	Av. # Steps used	Error
10^{-2}	Variable	20	0.0019
	Fixed		0.0070
10^{-3}	Variable	91	0.00051
	Fixed		0.0019
10^{-4}	Variable	417	0.00012
	Fixed		0.00043

in EPS mode, i.e. enforcing $E_d \leq \tau$, $E_s \leq \tau$ and $E_t \leq \tau$ and the results are shown in Table 4. The efficiency gains are comparable to those obtained in EPUS mode although the approximate tolerance proportionality of the EPUS simulations has been lost. These questions regarding the choice of Itô vs. Stratonovich drift and EPS vs. EPUT modes in the error controls will be considered further elsewhere.

Finally we consider the often-used test problem

$$dX = -\alpha(1 - X^2)\, dt + \beta(1 - X^2) \circ dW, \qquad X(0) = X_0 \tag{25}$$

with exact solution

$$X(t) = \frac{(1 + X_0)e^{-2\alpha t + 2\beta W(t)} + X_0 - 1}{(1 + X_0)e^{-2\alpha t + 2\beta W(t)} + 1 - X_0}.$$

Numerical results for the parameter values $\alpha = 1, \beta = 1.5$ are presented in Table 5. The adaptivity gains are far less striking, indeed the adaptive algorithm performs worse at large tolerances. The reason for this is that both the functions $\underline{f}(x)$ and $g(x)$ have critical points at $x = 0$ causing all three error estimates E_d, E_s and E_t to vanish. Thus, for trajectories that cross $x = 0$, the timesteps increase in the neighbourhood of this point degrading the global solution. However on trajectories that do not cross $x = 0$ the algorithm performs extremely well with an error reduction of approximately a factor of 8. While such a confluence of critical points is not likely to happen often in practice, this final example demonstrates the need for further refinement of the original algorithm.

Table 5: Numerical results for Test Problem 2 using EPS error control.

τ	Method	Av. # Steps used	Error
10^{-2}	Variable	80	0.066
	Fixed		0.029
10^{-3}	Variable	694	0.0049
	Fixed		0.0057
10^{-4}	Variable	6512	0.00045
	Fixed		0.00064

6 Conclusions

The algorithmic approach developed above can be generalized in various ways. The crucial starting point is the use of error estimates that are easily inverted to provide explicit bounds dependent upon h and ΔW — the EPUS/EPUT control used here is just one of several possibilities. These bounds then directly lead to the use of first-exit-times as optimal timesteps with no possibility for stepsize rejections. The density functions across the boundaries of such compact regions are known in various cases (see, for example, [22, 23]), the rectangular one used in Section 4 being the simplest.

There are certain computational issues that have not been addressed here but which may significantly affect the efficiency of the algorithm in practice. For example, it may not be necessary to update the error estimates on every timestep and the use of finite differences to approximate $\underline{f}$ would also reduce the number of function evaluations needed. Other factors will depend upon hardware and software considerations. The look-up tables could be replaced by approximating functions if it was faster to evaluate them than to access the table. Or if parallel processors are available then one processor could be dedicated to the generation of the exit-times while the other updated the solution and the error estimates.

This paper has only considered adaptivity for a strong-order 1 method applied to SDEs forced by a scalar Wiener process. However, the underlying concepts are not limited to this case. The use of higher-order methods and/or multiple stochastic processes requires the calculation of higher-order stochastic integrals such as J_{10} and J_{01} (except for SDEs satisfying certain commutativity conditions) which significantly increases the computational

cost of each timestep. A major problem with trying to adaptively change timesteps in this general case is that whenever timestep rejections occur all the stochastic integrals used to update the solution have to be recomputed over the two resulting sub-intervals, whilst respecting their joint conditional probability distributions. This increases the cost and complexity of the algorithm even further but ceases to be an issue if stepsize rejections cannot occur. If multi-dimensional stochastic forcing is present it may then be possible to combine explicit error estimates with results concerning the distribution of multi-dimensional Brownian motions exiting higher-dimensional boundaries to construct viable adaptive algorithms.

References

[1] T. W. Anderson. A modification of the sequential probability ratio test to reduce the sample size. *Ann. Math. Stat.*, 31:165–197, 1960.

[2] K. Burrage, P.M. Burrage, and T. Mitsui. Numerical solutions of stochastic differential equations—implementation and stability issues. *J. Comput. Appl. Math.*, 125(1-2):171–182, 2000.

[3] P. M. Burrage and K. Burrage. A variable stepsize implementation for stochastic differential equations. *SIAM J. Sci. Comput.*, 24(3):848–864, 2002.

[4] P. M. Burrage, R. Herdiana, and K. Burrage. Adaptive stepsize based on control theory for stochastic differential equations. *Journal of Computational and Applied Mathematics*, 170:317–336, 2004.

[5] J.G. Gaines and T.J. Lyons. Variable step size control in the numerical solution of stochastic differential equations. *SIAM J. Appl. Math.*, 57(5):1455–1484, 1997.

[6] K. Gustafsson. Control-theoretic techniques for stepsize selection in explicit Runge-Kutta methods. *ACM Trans. Math. Software*, 17(4):533–554, 1991.

[7] E. Hairer, S.P. Nørsett, and G. Wanner. *Solving ordinary differential equations I. Nonstiff problems*. Springer-Verlag, Berlin, second edition, 1993.

[8] W. J. Hall. The distribution of Brownian motion on linear stopping boundaries. *Sequential Analysis*, 16:345–352, 1997.

[9] N. Hofmann, T. Müller-Gronbach, and K. Ritter. Optimal approximation of stochastic differential equations by adaptive step-size control. *Math. Comp.*, 69(231):1017–1034, 2000.

[10] N. Hofmann, T. Müller-Gronbach, and K. Ritter. The optimal discretization of SDEs. *J. Complexity*, 17:117–153, 2001.

[11] N. Hofmann, T. Müller-Gronbach, and K. Ritter. Optimal uniform approximations of systems of SDEs. *Ann. Appl. Prob.*, 12:664–690, 2002.

[12] K. M. Jansons and G. D. Lythe. Efficient numerical solution of stochastic diffrential equations using exponential timestepping. *J. Stat. Phys.*, 100:1097–1109, 2000.

[13] K. M. Jansons and G. D. Lythe. Exponential timestepping with boundary test for stochastic differential equations. *SIAM J. Comp. Sci.*, 24:1809–1822, 2003.

[14] Ioannis Karatzas and Steven E. Shreve. *Brownian Motion and Stochastic Calculus*. Springer-Verlag, second edition, 1991.

[15] P.E. Kloeden and E. Platen. *Numerical solution of stochastic differential equations*. Springer-Verlag, Berlin, 1992.

[16] H. Lamba. Dynamical systems and adaptive time-stepping in ODE solvers. *BIT*, 40:314–335, 2000.

[17] H. Lamba. An adaptive timestepping algorithm for stochastic differential equations. *J. Comp. Appl. Math.*, 161:417–430, 2003.

[18] H. Lamba. The effective stability of adaptive timestepping ODE solvers. *SIAM J. Num. Anal.*, 43:1950–1968, 2005.

[19] H. Lamba, J. Mattingly, and A.M. Stuart. An adaptive euler-maruyama scheme for stochastic differential equations: convergence and stability. *IMA J. Num. Anal.*, 27:479–506, 2006.

[20] H. Lamba and T. Seaman. Mean-square stability properties of an adaptive time-stepping SDE solver. *J. Comp. Appl. Math.*, 194(2):245–254, 2006.

[21] S. Mauthner. Step size control in the numerical solution of stochastic differential equations. *J. Comput. Appl. Math.*, 100(1):93–109, 1998.

[22] A. Novikov, V. Frishling, and N. Kordzakhia. Approximations of boundary crossing probabilities for a Brownian motion. *J. Appl. Prob.*, 36:1019–1030, 1999.

[23] K. Pötzelberger and L. Wang. Boundary crossing probability for Brownian motion. *J. Appl. Prob.*, 38:152–164, 2001.

[24] W. Römisch and R. Winkler. Stochastic DAEs in circuit simulation. In *Modeling, simulation, and optimization of integrated circuits (Oberwolfach, 2001)*, volume 146 of *Internat. Ser. Numer. Math.*, pages 303–318. Birkhäuser, Basel, 2003.

[25] Y. Saito and T. Mitsui. Stability analysis of numerical schemes for stochastic differential equations. *SIAM J. Numer. Anal.*, 33(6):2254–2267, 1996.

[26] L.F. Shampine. *Numerical Solution of Ordinary Differential Equations*. Chapman and Hall, New York, 1994.

[27] A.M. Stuart. Probabilistic and deterministic convergence proofs for software for initial value problems. *Numerical Algorithms*, 14:227–260, 1997.

[28] A.M. Stuart and A.R. Humphries. The essential stability of local error control for dynamical systems. *SIAM J. Num. Anal.*, 32:1940–1971, 1995.

[29] A. Szepessy, R. Tempone, and G.E. Zouraris. Adaptive weak approximation of stochastic differential equations. *Comm. Pure Appl. Math.*, 54(10):1169–1214, 2001.