Numerical Methods For Stochastic Differential Equations

Samuel Eure

MATH 361S, Final Paper

Duke University

December 9, 2018

Abstract

This report provides an introduction to common numerical methods used to simulate and solve stochastic differential equations (SDEs). Relevant background information is provided regarding stochastic processes, however it is assumed that the reader possesses some familiarity with random variables and numerical ordinary differential equations (ODEs). Numerical techniques for dealing with SDEs such as the Euler-Maruyama and Milstein methods are discussed and algorithms are provided for their implementation. Finally, an analysis of these methods' convergences is carried out by exploring the concept of error for SDEs through weak and strong convergence.

1 Introduction

Stochastic differential equations are key mathematical modeling tools which are applied to problems in fields ranging from epidemiology to finance. Indeed, some of the greatest advancements in the field of stochastic processes have been made while studying time dependent variables such as stock prices and the populations of species [Roy & Rau, 2017]. Although SDEs are useful in a multitude of ways, the magnitude of their applicability matches that of their complexity, and in order to fully comprehend and derive their solutions analytically, mastery of stochastic processes and ordinary differential equations is required. Nevertheless, with a working knowledge of random variables, probability distributions, and familiarity with basic numerical methods used to solve ODEs[1], the background required to successfully implement numerical methods for dealing with SDE is far less challenging. This report provides a few such methods, describes their key differences, and discusses how numerical methods of their nature are generally improved. In order to accomplish these goals, a brief overview of stochastic processes will first be provided, presenting necessary concepts and phenomenon for the successful implementation of numerical SDE solvers.

1.1 Stochastic Processes

Suppose X is a random variable drawn from some probability distribution Θ. Then, before the value of X is *realized* (i.e. X has been observed from Θ) X may take any value from the sample space of the Θ distribution. Now, consider the variable X_t such that $X_t \sim \Theta$, and now the value of X_t is indexed by time. Under these conditions, X_t is called

[1]It is expected the reader will have an understanding of probability theory at the level of an introductory undergraduate course, and should have some exposure to the Euler method for solving ODEs numerically.

a *continuous-time stochastic process* (CTSP), where each realization of X_t depends on what values are drawn from Θ over for different points $t \geq 0$ [Sauer, 2011]. Thus, X_t can be thought of a function of t whose values map from real time values to the outcome space of Θ.

One canonical CTSP is the *Wiener process* (represented at $\mathcal{W}_t$ below) which will be used throughout this report and which satisfies these main properties [Roy & Rau, 2017]:

1. $\mathcal{W}_t \sim \mathcal{N}(0, t), \quad \forall t \geq 0$

2. If $[a, b]$ and $[c, d]$ are disjoint time intervals, $\mathcal{W}_b - \mathcal{W}_a$ is i.i.d. from $\mathcal{W}_d - \mathcal{W}_c$

3. The Wiener process $\mathcal{W}_t$ can be represented as a continuous path.

The Wiener process is a well established model for Brownian motion and the two terms are often used interchangably. If a Wiener process of N time steps from time 0 to time T is desired, the following algorithm can be implemented numerically [Higham, 2001] – a few realizations of which are displayed in **Figure 1**. Note the variation in the paths for different realizations.

Algorithm 1 N Step Wiener Process Simulation

Input[**N**], **Return**[$\mathcal{W}$]
$X \sim \mathcal{N}(0, 1)$
$T = 1; h = T/N$
$d\mathcal{W}_1 = x_0\sqrt{h}$
$\mathcal{W}_1 = d\mathcal{W}_1$
for $k = 2 : n$ **do**
    $d\mathcal{W}_k = x_k\sqrt{h}$
    $\mathcal{W}_k = \mathcal{W}_{k-1} + d\mathcal{W}(k)$
end for

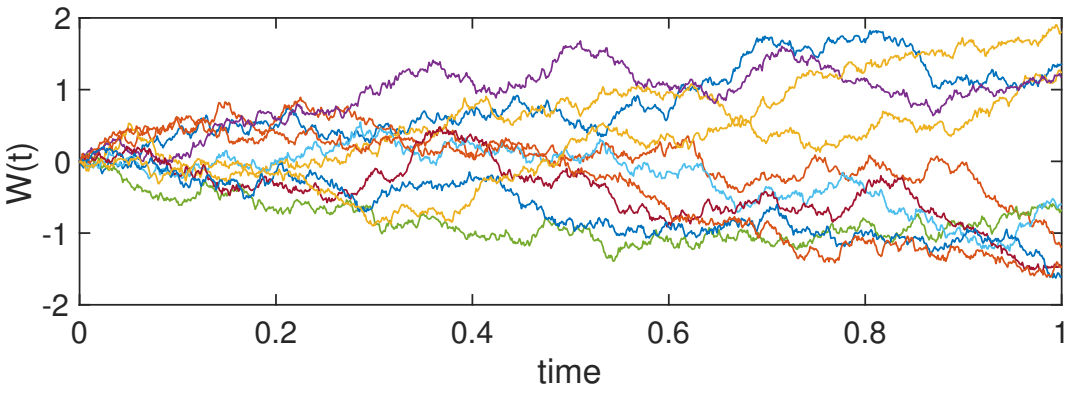

Figure 1: Ten Simulated Wiener Processes ($N = 1000$)

1.2 Stochastic Differential Equations

While studying ODEs, when the derivative of a function $y(t)$ can be represented as a function $f(t, y(t))$, equations such as

$$dy = f(t, y(t))dt$$

often arise. For SDEs, if $X(t)$ is the stochastic analogue of $y(t)$, equations of the form

$$dX_t = a(X_t, t)dt + b(X_t, t)d\mathcal{W}_t \tag{1.1}$$

similarly arise where a and b are functions of t and X_t, but instead of representing a derivative of $X(t)$, a and b quantify the relationship between dt (respectively $d\mathcal{W}_t$) and X_t. In fact, the paths represented by stochastic processes such as the Wiener process are continuous, but nowhere differentiable[2] [Roy & Rau, 2017]. As a result of this, it is more appropriate to

[2]This is the case for stochastic processes which are said to have a component which takes a *random walk*, where the direction towards which $X(t)$ is traveling (i.e. its derivative) "jumps" in direction between any two time intervals.

define stochastic differential equations as equations of the form

$$X(T) = X_0 + \int_0^T a(X_t, t)dt + \int_0^T b(X_t, t)d\mathcal{W}_t \tag{1.2}$$

whose solutions are CTSPs for some starting condition X_0. Unlike ODEs (which have a unique solution given any initial condition), stochastic differential equations have different solutions for different realized stochastic processes − as previously seen in **Figure 1** for the Wiener process.

In equation (1.2), the significance of $\int_0^T a(X_t, t)dt$ is quite clear; however, what integrating b with respect to $d\mathcal{W}$ means is not so obvious. To understand this portion of the SDE, it helps to recall that an integral can be though of as taking an infinite Riemann sums:

$$\int_0^T a(t, X)dt = \lim_{\Delta t \to 0} \sum_{i=1}^{n} a(t_{i-1}, X(t_{i-1}))\Delta t \tag{1.3}$$

Similarly, an equation such as this can be formed with the latter portion of the SDE:

$$\int_0^T b(t, X)d\mathcal{W}_t = \lim_{\Delta t \to 0} \sum_{i=1}^{n} b(t_{i-1}, X(t_{i-1})\Delta\mathcal{W}_i \tag{1.4}$$

where $\Delta\mathcal{W}_i \sim \mathcal{N}(0, \Delta t)$. In a way, this integral quantifies the amount of *white noise* involved in the SDE, and it may be though of as such for the purposes of this report. Just as the integral form of SDEs may be approximated by taking a sequence of Riemman sums, equation (1.1) can be approximated as

$$\Delta X_t = a(t, X(t))\Delta t + b(t, X(t))\Delta\mathcal{W}_t \tag{1.5}$$

which can of course be implemented numerically quite easily. The idea of approximating the SDE in this fashion quite naturally leads into the Euler-Maruyama method for solving SDEs, which is discussed in the following section.

2 The Euler-Maruyama Method for Solving SDEs

Arguments used to obtain equations such as (1.5) are the cornerstone of developing numerical methods for solving SDEs, and the numerical methods discussed here will involve approximations such as $dX_t \approx \Delta X_t = X_t - X_0$ and $d\mathcal{W} \approx \Delta \mathcal{W}_i \sim \mathcal{N}(0, \Delta t)$ for some discrete, determined time change Δt. In this way, the realized path of $X(t)$ will be approximated explicitly in order to obtain a numerical solution. As aforementioned in the previous section, the Euler-Maruyama method is the first of such method where this general technique is used.

The Euler-Maruyama (EM) method is the stochastic analogue of the Euler method for solving ordinary differential equations. For SDEs such as (1.1) starting at $X_{t_0} = x_0$, the Euler-Maruyama method takes the form of:

$$X_{t_{n+1}} = X_{t_n} + a(X_{t_n}, x)\Delta t + b(X_{t_n}, t_n)\Delta W \tag{2.1}$$

where $\Delta X_t = X_{t_{n+1}} - X_{t_n}$, $\Delta t = t_{n+1} - t_n$ and $\Delta W = W(t_{n+1}) - W(t_n)$. This method approximates equations (1.3) and (1.4) for a predetermined time-step. To further illustrate how EM may be applied to SDEs, consider an equation of the form

$$dX(t) = \mu X_t dt + \sigma X_t d\mathcal{W}_t \tag{2.2}$$

where μ and σ are real constants and $X(t)$ is a stochastic process. An SDE of this form is said to follow a *geometric brownian path*, since its solutions are comprised of a realized Brownian path along with a *drift* term [Rosa, 2016]. If a solution is desired for equation (2.2) on the interval [0,T] with N iterations, the Euler-Maruyama method can easily be used to find an approximate solution to this equation, as implemented below where h (i.e. Δt) is the time-step.

Algorithm 2 Euler-Maruyama Implementation

Input$[\sigma, \mu, \mathbf{x_0}, \mathbf{T}, \mathbf{N}]$, **Return**$[\mathbf{X_t}]$
$h = T/N$
$\mathcal{W} = N$ step Wiener process from **Algorithm 1**
$X =$ array of length N
$X(0) = x_0$
for $k = 0, 1, 2, ...N$ **do**
    $X(k+1) = X(k) + \mu X(k)h + \sigma X(k)(\mathcal{W}(k+1) - \mathcal{W}(k))$
end for

Equation (2.2) can also be solved analytically quite easily [Oksendal, 1998], and is known to have the exact solution of

$$X(t) = x_0 e^{\sigma \mathcal{W}_t + (\mu - \sigma^2/2)t} \tag{2.3}$$

for some initial condition x_0. With this knowledge, the Euler-Maruyama's accuracy can be visualized by providing the EM method's approximation to equation (2.3) and equation (2.3) the same realization of the N step Wiener process and plotting them alongside each other[3]. For example, consider solutions to the stochastic differential equation

$$dX(t) = 0.8X_t dt + 0.3X_t d\mathcal{W}_t \tag{2.4}$$

[3]Recall that although there is an exact solution to equation 2.2, there is not a unique solution, as the exact solution is dependent on the realization of $d\mathcal{W}$ between each change in time.

starting at $x_0 = 100$ over the time interval $t \in [0, 1]$. Using **Algorithm 2**, one realization of the solutions may be obtained, as shown in Figure 2.

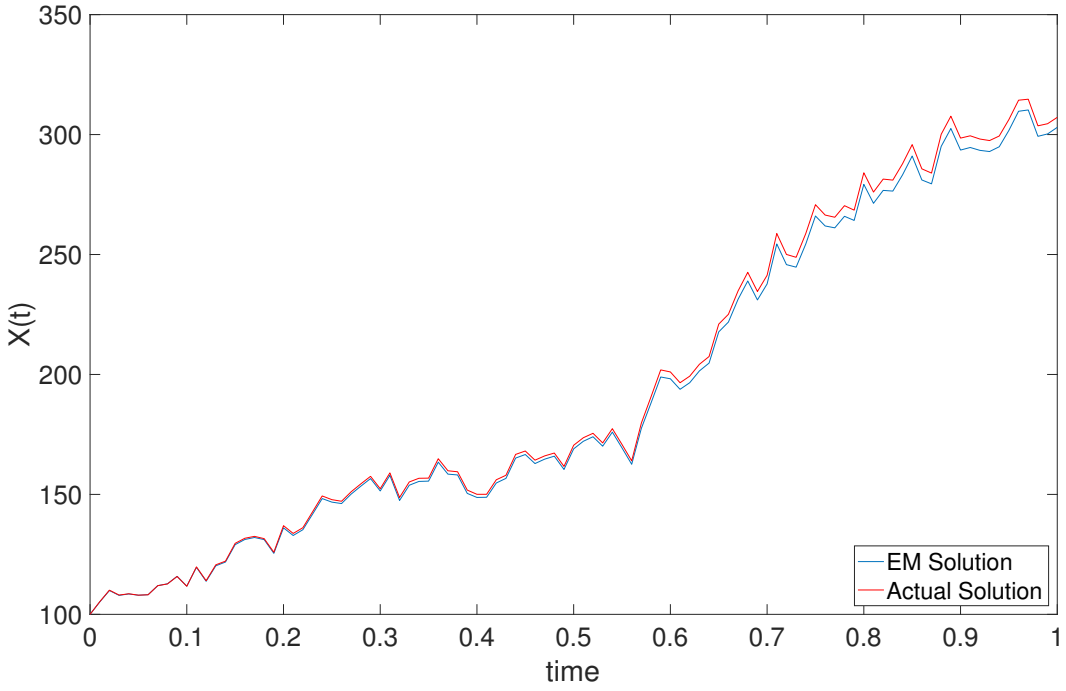

Figure 2: A realized solutions to $dX(t) = 0.8X_t dt + 0.3X_t d\mathcal{W}_t$ from the EM Method ($N = 100$).

As shown above, the EM method seems to approximate the SDEs quite well. But how well? To answer this question, the concept of error for numerical SDEs be explored.

3 Error Analysis

For ODEs, the concept of error for a numerical method is quite intuitive: how large the difference is between the value of the numerical solution to the actual, unique solution of the ODE. With SDEs, the idea of error is not so simple. For one, there does not exist a

unique solution of an SDE, only realized solutions. However, the idea of comparing numerical approximations to an *expected* path of the exact solution is something conceptually pleasing. Since stochastic processes are continuous time dependent random variables, there does in fact exist such a path which the exact solution of an SDEs is expected to take. For example, if $S(t)$ is the exact solution of an SDE, then $\mathbb{E}[S(t_n)]$ is the expected value of $S(t_n)$ at some time $t_n \in [0, T]$. Thus, if $X(t)$ is the realized numerical approximation of $S(t)$, the concept of error at a particular time t can be evaluated as $\mathbb{E}\left[|S(t) - X(t)|\right]$, or as $|\,\mathbb{E}\left[S(t)\right] - \mathbb{E}\left[X(t)\right]\,|$.

3.1 Strong Convergence

A numerical approximation $X(t)$ is said to *converge strongly* to a stochastic process $S(t)$ at time T if

$$\lim_{\Delta t \to 0} \mathbb{E}\,|S(T) - X(T)| = 0$$

Similar to the idea of global truncation error for ODEs, the concept of strong convergence is a statement about the greatest error of $\mathbb{E}\left[|S(t_n) - X(t_n)|\right]$ for any $t_n \in [0, T]$. To be more explicit, if $h \geq \max_{\forall t_n \in [0,T]} |(t_{n+1} - t_n)|$, then if

$$\mathbb{E}\left[|S(t_n) - X(t_n)|\right] = O(h^p), \qquad \forall t_n \in [0, T] \tag{3.1}$$

$X(t)$ is said to have a strong order of convergence of p. This is a statement about the *mean of the error* at any point t along the entire path of the solution $S(t)$.

Surprisingly, even the order of convergence for the Euler method for ODEs is 1, the order of strong convergence for the EM method is only 1/2 [Higham, 2001]. To show this, if

$\epsilon(t) = |S(t) - X(t)|$, then $\epsilon(t) = O(h^{1/2})$, which implies $\epsilon(t) \approx Ch^{1/2}$ for some finite constant C. Using the fact that $\log(\epsilon(t)) \approx \log(C) + \frac{1}{2}\log(h)$, a linear trend should exist between the $\log(\epsilon(t))$ and $\log(h)$. By averaging 1000 numerically generated realizations of $\epsilon(T)$ for equation (2.4), and then repeating this procedure for increasing values of Δt, Figure 3 is obtained.

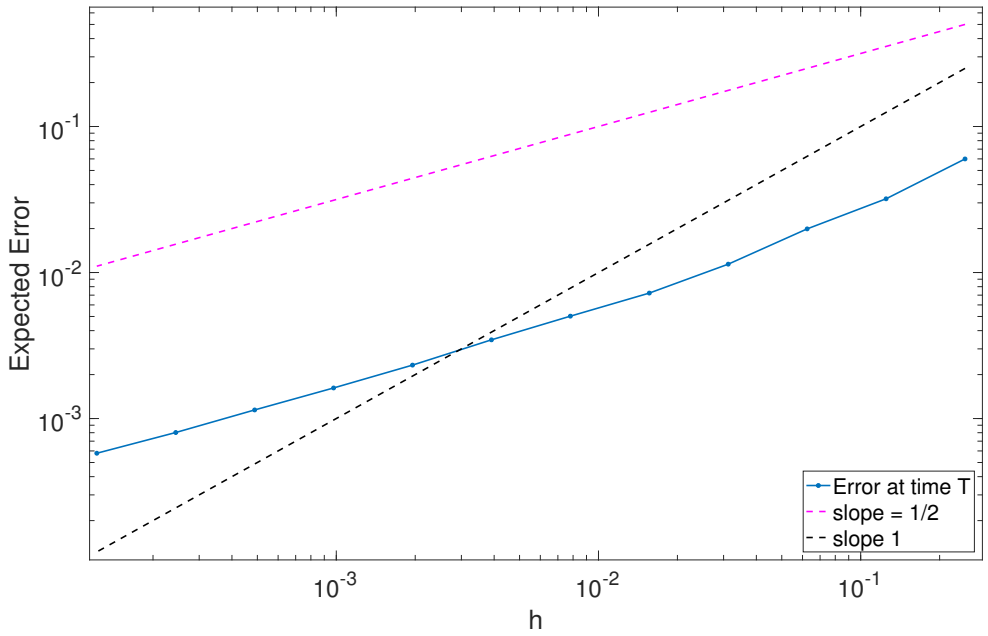

Figure 3: Strong Convergence of EM for equation (2.4) for 1000 samples

It is clear from Figure 3 that the predicted linear trend with slope 1/2 is present for the strong error of the Euler-Maruyama method[4]. The significance of this is that as $h \to 0$, the expected error at any time along the interval $[0, T]$ will go to zero[5]. Sometimes a less stringent evaluation of error is desired − or at least sufficient − for a particular problem at hand, which is the error between the expected values of $S(t)$ and $X(t)$. This idea of convergence is more

[4]The slope of 1/2 in Figure 3 is given for comparison and the slope of 1 is provided for contrast.

[5]It is for this reason that the error is referred to as *strong*, since it states that each of the realized numerical approximations will converge to the true solution as $h \to 0$.

relaxed but less relatable, and is appropriately denoted as *weak convergence*.

3.2 Weak Convergence

If strong convergence is thought of as the mean of the errors at any particular time, weak convergence can be thought of as *the error of the means* at any particular time along our solution's interval $[0, T]$. To be precise, if $h \geq \max_{\forall t_n \in [0,T]} |(t_{n+1} - t_n)|$, and if $\Psi(\cdot)$ satisfies smoothness and polynomial growth conditions, then if

$$|\,\mathbb{E}\,[\Psi(S(t))] - \mathbb{E}\,[\Psi(X(t))]\,| = O(h^q), \qquad \forall t_n \in [0, T] \tag{3.2}$$

X_t is said to have a *weak order of convergence* of q [Sauer, 2011]. This type of convergence is referred to as "weak" since it is a statement about the error of the expected values $S(t)$ and $X(t)$ and does not constrain the maximum error between any particular realizations of $S(t)$ and $X(t)$. To check the order of convergences for the stochastic processes in this report, it suffices to do so for $\Psi(X(t)) = X(t)$ [Higham, 2001]. Just as $\mathbb{E}\,[\epsilon(T)]$ was approximated with 1000 realizations of $\epsilon(T)$, $\mathbb{E}\,[S(t)]$ and $\mathbb{E}\,[X(t)]$ may be calculated similarly. By doing so, the weak order of convergence for equation (2.4) for EM can be visualized, as Figure 4 displays.

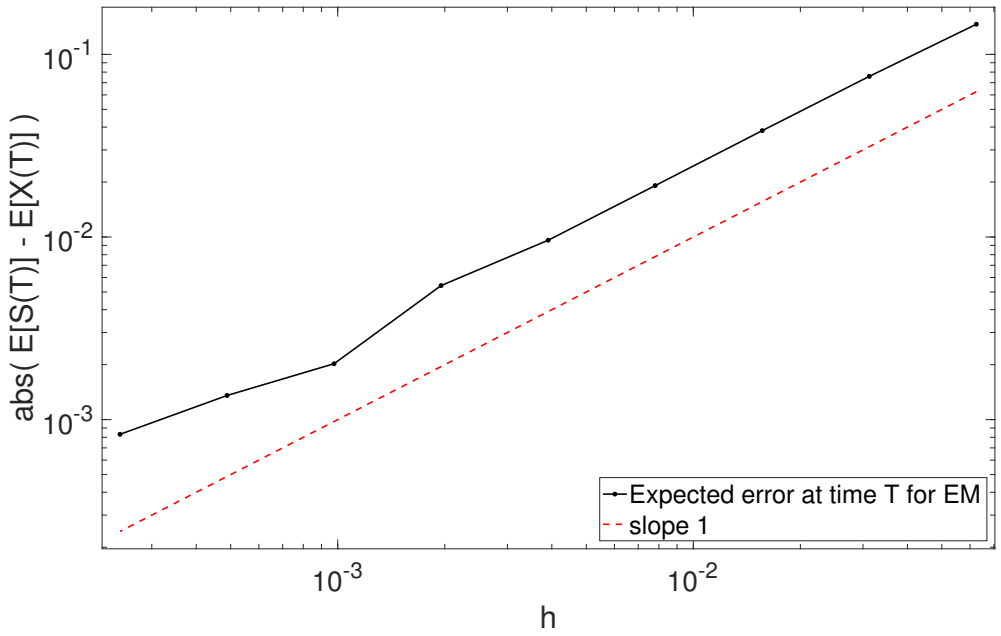

Figure 4: Weak Convergence of EM for equation (2.4) for $400,000$ samples

As the reader may have expected, the weak order of convergence for EM is greater than that of its strong convergence: it is 1. This is generally the case for numerical methods used to solve SDEs, but not always. Having presented the Euler-Maruyama method and discussed its convergences, a natural question to ask is "how can the numerical approximation be improved?" To proceed in answering this question, additional concepts related to stochastic processes must first be introduced, starting with one of the most significant theories in all of stochastic calculus: Ito's Lemma.

Higher Order Numerical Methods

Many numerical methods used to improve the Euler method for solving ODEs rely on Taylor expanding the equation of interest around a discrete point in order to account for

truncation errors. Similarly, numerical methods used for solving SDEs involve such a practice, though the necessity behind doing so is somewhat different, as discussed in this section.

Ito's Lemma

For some function $f(t, X(t)) \in \mathcal{C}''$, where t = time and $X(t)$ is a CTSP, $df(t, X(t)$ can be approximated through a Taylor expansion as

$$df = \frac{\partial f}{\partial t}dt + \frac{\partial f}{\partial X}dX +$$

$$\frac{1}{2}\left(\frac{\partial^2 f}{\partial X^2}dX^2 + 2\frac{\partial^2 f}{\partial X \partial t}dX\,dt + \frac{\partial^2 f}{\partial t^2}dt^2\right) + ...$$

If $X(t)$ satisfies $dX = \mu dt + \sigma d\mathcal{W}_t$, then this substitution can be made for dX, resulting in

$$df = \frac{\partial f}{\partial t}dt + \frac{\partial f}{\partial X}(\mu dt + \sigma d\mathcal{W}_t) +$$

$$\frac{1}{2}\left(\frac{\partial^2 f}{\partial X^2}(\mu dt + \sigma d\mathcal{W}_t)^2 + 2\frac{\partial^2 f}{\partial X \partial t}(\mu dt + \sigma d\mathcal{W}_t)dt + \frac{\partial^2 f}{\partial t^2}dt^2\right) + ...$$

It turns out that for a Wiener process $\mathcal{W}$, the variance of $\mathcal{W}$ grows quadratically with respect to time (i.e. $d\mathcal{W}^2 \approx dt$) [Roy & Rau, 2017]. Thus, $d\mathcal{W}^2$ approaches zero slower than dt^2 and $dtd\mathcal{W}$ as $h \to 0$. As a result, a better approximation of df which accounts for $d\mathcal{W}^2 \approx dt$ is

$$df = \frac{\partial f}{\partial t}dt + \frac{\partial f}{\partial X}(\mu dt + \sigma d\mathcal{W}_t) + \frac{1}{2}\frac{\partial^2 f}{\partial X^2}\sigma^2 dt. \tag{3.3}$$

This form of $df(t, X_t)$ for stochastic functions is known as **Ito's Lemma**, and is fundamental for the study of stochastic calculus and SDEs. By using Ito's Lemma, corrections can be

made for the approximation of dX used for the Euler-Maruyama method; such methods involve including higher order terms to equation (1.5). One such method which will be discussed in this report is the Milstein method.

3.3 The Milstein Method

For equations of the form (2.2), using Ito's Lemma, it can be shown that :

$$d(ln(X_t)) = \left(\mu - \frac{1}{2}\sigma^2\right) dt + \sigma d\mathcal{W}_t,$$

which can be put in the form of

$$X_{t+dt} = X_t exp\left\{\int_t^{t+dt} \left(\mu - \frac{1}{2}\sigma^2\right) dt + \int_t^{t+dt} \sigma d\mathcal{W}\right\}$$

and, for small Δt, be approximated as

$$X_{t+dt} \approx X_t \left(1 + \mu\Delta t - \frac{1}{2}\sigma^2\Delta t + \sigma\Delta\mathcal{W}_t + \frac{1}{2}\sigma^2(\Delta\mathcal{W}_t)^2\right)$$

$$\approx X_t + \mu X_t\Delta t + \sigma X_t\Delta\mathcal{W}_t + \frac{1}{2}\sigma^2 X_t\left(\Delta\mathcal{W}^2 - \Delta t\right). \tag{3.4}$$

Equation (3.4) is the Milstein method's approximation for equation (2.2). More generally, for any SDE with the form of equation (1.1), the Milstein Method takes the form of

$$dX_t = a(X_t, t)dt + b(X_t, t)d\mathcal{W}_t + \frac{1}{2}b(X_t, t)\frac{\partial b}{\partial X}(X_t, t)\left(d\mathcal{W}^2 - dt\right).$$

Notice that the Milstein method is an extention of the Euler-Maruyama method which makes a correction for the quadratic variation of $d\mathcal{W}$. Continuing with equation (2.2) for geometric Brownian motion and using the fact that $\dfrac{\partial b}{\partial X} = \sigma$, the Milstein method can be implemented as shown in **Algorithm 3**.

Algorithm 3 Milstein Implementation

Input$[\sigma, \mu, \mathbf{x_0}, \mathbf{T}, \mathbf{N}]$, **Return**$[\mathbf{X_t}]$
$h = T/N$
$\mathcal{W} = N$ step Wiener process from **Algorithm 1**
$d\mathcal{W}_k = (\mathcal{W}(k+1) - \mathcal{W}(k))$
$X =$ array of length N
$X(0) = x_0$
for $k = 0, 1, 2, ...N$ **do**
$d\mathcal{W} = \mathcal{W}(k+1) - \mathcal{W}(k)$
$X(k+1) = X(k) + \mu X(k)h + \sigma X(k)(d\mathcal{W})$
$X(k+1) = X(k+1) + X(k)\frac{1}{2}\sigma^2\left((d\mathcal{W})^2 - h\right)$
end for

To directly visualize the effects of this correction, numerical solutions obtained from the Milstein method as well as the Euler-Maruyama method can be plotted alongside one another, as shown in Figure 5. Note how for equations which have σ values close to μ, the Milstein approximation follows the exact solution much more closely. To understand just how much the accuracy is expected to increase by, the methods used to show the strong and weak convergences of the Euler-Maruyama method can be repeated for the Milstein method, as shown in Figures 6 & 7, respectively.

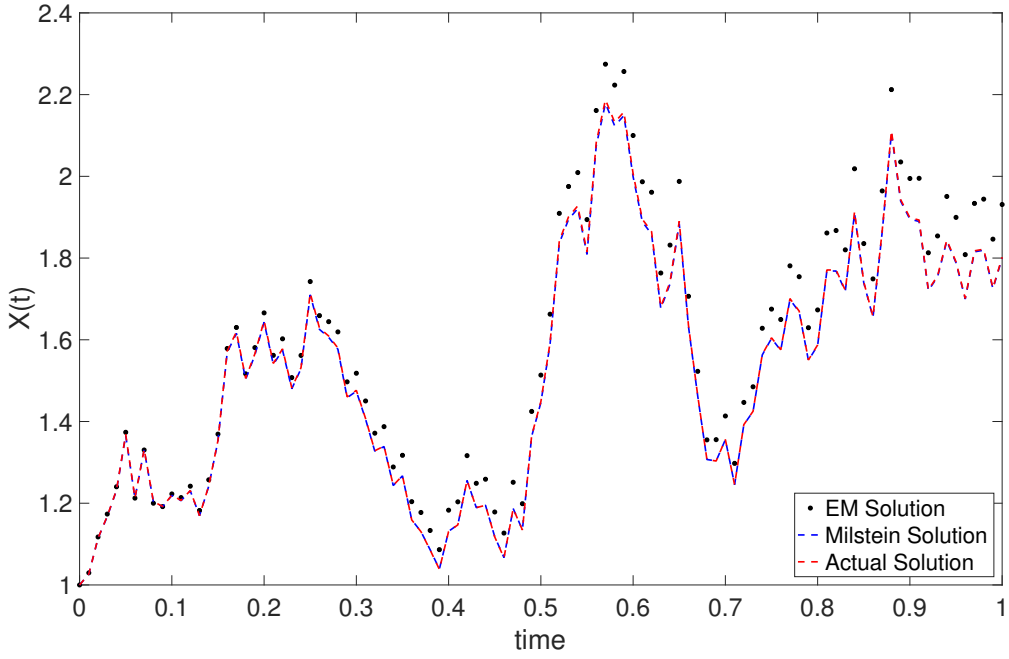

Figure 5: A realized solutions to $dX(t) = 0.8X_t dt + 0.8X_t d\mathcal{W}_t$ from the Milstein Method ($N = 100, X_0 = 4$).

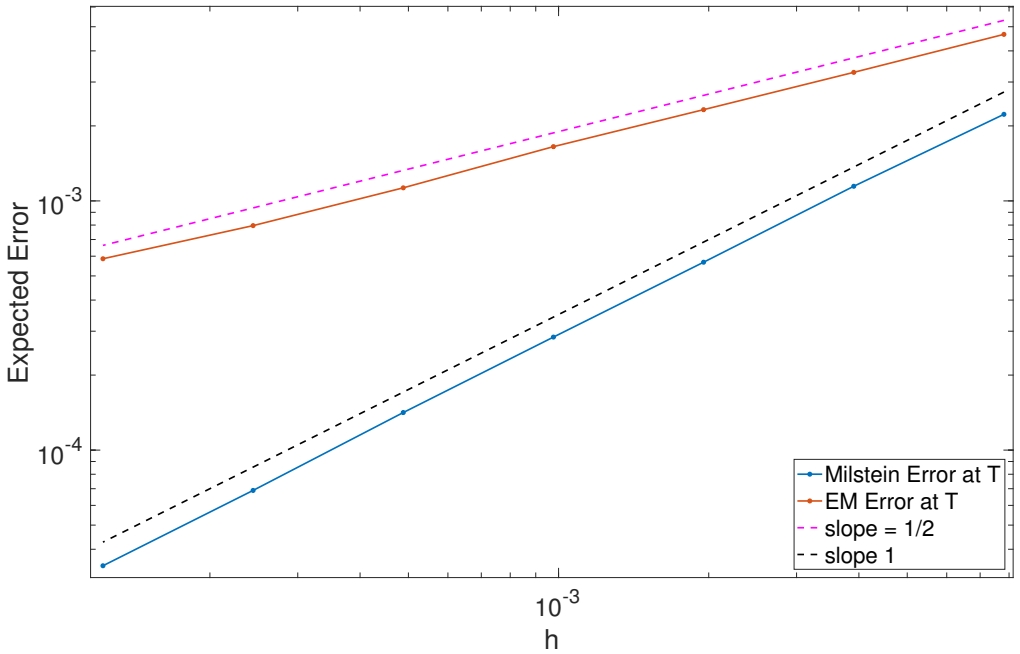

Figure 6: Strong Convergence of the Milstein Method on $dX(t) = 0.8X_t dt + 0.3X_t d\mathcal{W}_t$ (1000 Samples, $X_0 = 100$).

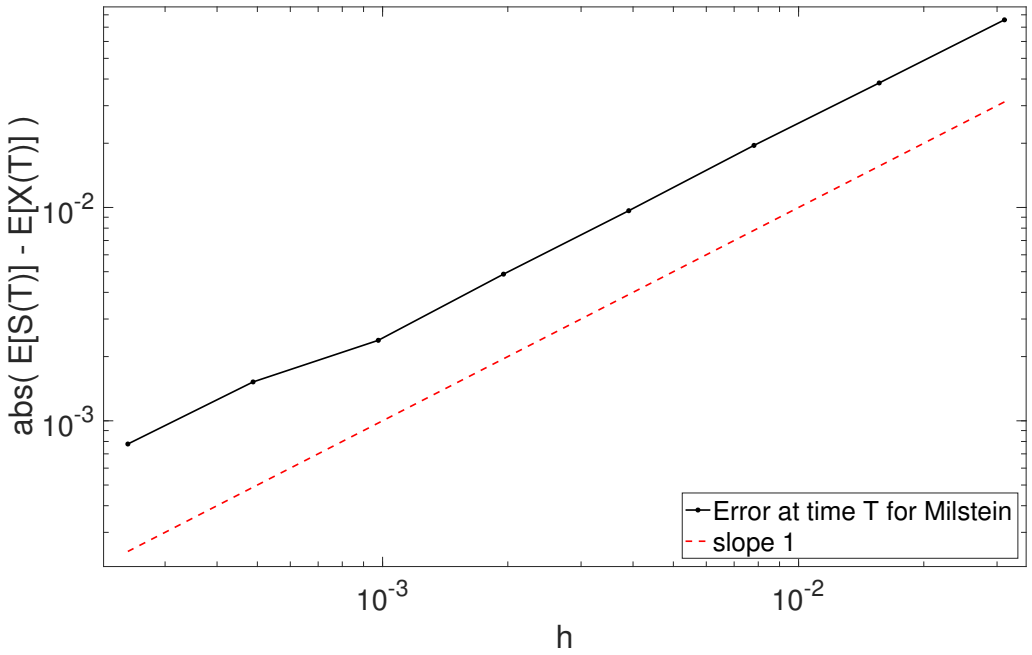

Figure 7: Weak Convergence of the Milstein Method on $dX(t) = 0.8X_t dt + 0.3X_t d\mathcal{W}_t$ $(1,000,000$ Samples, $X_0 = 1)$.

As shown in Figure 6, the Milstein method has a strong convergence of 1 − as opposed to 1/2 by the EM method − however the weak order of convergence remains 1. It is possible to increase both the strong and weak order of convergence by expanding out the Taylor series of equations of the form (2.2) even further and applying Ito's Lemma. This is the basis of other higher order methods for solving SDEs numerically such as the Runge-Kutta method [Rosa, 2016].

4 Summary

Numerical methods used to solve stochastic differential equations follow many of the same techniques as those used to solve ordinary differential equations numerically, such as the Euler method. Many of these methods require dealing with differential equation which contain a

Wiener process component that adds randomness to the solutions of such equations. This addition of randomness results in solutions which are not unique, and forces the evaluation of error for numerical methods to rely on concepts from the expectation of random variables − such as strong and weak convergences. Such errors for numerical methods can be reduced by resorting to stochastic calculus (mainly Ito's Lemma) and Taylor expanding out formulas of interest to obtain more accuracy. This report has intended to provide a brief introduction to formulas and algorithms use to numerically solve stochastic differential equations following a geometric Brownian motion and may be used as a primer for the study of numerical SDEs.

References

[Sauer, 2011] Sauer, Timothy. *"Numerical Solutions of Stochastic Differential Equations in Finance"*, Department of Mathematics at George Mason University (2011)

[Roy & Rau, 2017] Roy, Debashish & Rao, G. Visweswara. *"Stochastic Dynamics, Filtering and Optimization"*, Cambridge University Press (2017)

[Higham, 2001] Higham, Desmond. *"An Algorithmic Introduction to Numerical Simulations of Stochastic Differential Equations"*, Society for Industrial and Applied Mathematics, Vol. 43, No. 3, pp. 525-546 (2001)

[Oksendal, 1998] Oksendal, Bernt. *"Stochastic Differential Equations: An Inroduction with Applications",* Springer, Fifth Edition (1998)

[Rosa, 2016] Rosa, M. *"Numerical Methods for Stochastic Differential Equations with Applications to Finance"*, University of Lisbon, Portugal (2016)