Numerical methods for stochastic differential equations

Written by: David F. Anderson

Written for: Math 635, Spring 2012

April 27, 2012

We suppose that X_t is a solution to the one dimensional SDE

$$dX_t = a(t, X_t)dt + b(t, X_t)dB_t,$$

or

$$X_t = X_0 + \int_0^t a(s, X_s)ds + \int_0^t b(s, X_s)dB_s.$$

Following our existence and uniqueness results, we will assume that the coefficients satisfy a Lipschitz condition

$$|a(t, x) - a(t, y)|^2 + |b(t, x) - b(t, y)|^2 \leq K|x - y|^2$$

and the growth condition (used for existence)

$$|a(t, x)|^2 + |b(t, x)|^2 \leq K(1 + |x|^2) \quad \forall t.$$

Later we will discuss multi-dimensional SDEs, but this is a fine starting point to get the basic ideas.

Why do we want numerical methods? There are two reasons that come to mind instantly:

1. Perhaps we simply want to visualize a realization of the SDE to gain insight into the process.

2. Perhaps we want to know the value of

$$\mathbb{E}f(X_t), \quad \text{or maybe} \quad \mathbb{E}\left[\int_0^t f(X_s)ds\right], \tag{1}$$

where f is some function, but we have been unsuccessful in solving the SDE. Here Monte Carlo will be useful.

Monte Carlo

Both problems in (1) can be stated in terms of solving for $\mathbb{E}Z$ where Z is some random variable of interest. Suppose that we can generate independent realizations of Z using a computer (for example, maybe $Z = f(U)$, where U is a uniform$[0, 1]$ random variable[1].) Then, we could define *estimator* of $\mathbb{E}Z$ to be

$$\hat{\mu}_n = \frac{1}{n}\sum_{i=1}^{n} Z_{[i]},$$

[1]Later we will see that almost all interesting random variables can be viewed as a slight generalization of this setup

where $Z_{[i]}$ is the ith independent realization. We then have that

$$\mathbb{E}\hat{\mu}_n = \mathbb{E}Z,$$

and we also know from the law of large numbers (and a few minor conditions on the moments of Z),

$$\mu_n \to \mathbb{E}Z, \text{ as } n \to \infty, \quad a.s.$$

However, it is typically not enough to know that the estimator simply converges. Usually, one would like to build *confidence intervals*, which can be done by using the central limit theorem. Briefly, and denoting the unknown $\mathbb{E}Z = \mu$ and $Var(Z) = \sigma^2$, we know that

$$\frac{\sum_{i=1}^n Z_{[i]} - n\mu}{\sigma\sqrt{n}} \Rightarrow N(0,1),$$

where the convergence is in distribution. Also, for any $a > 0$, we have that

$$\begin{aligned} P\left(-a \leq \frac{\sum_{i=1}^n Z_{[i]} - n\mu}{\sigma\sqrt{n}} \leq a\right) &= P\left(-a\sigma\sqrt{n} \leq \sum_{i=1}^n Z_{[i]} - n\mu \leq a\sigma\sqrt{n}\right) \\ &= P\left(-a\sigma\sqrt{n} - \frac{1}{n}\sum_{i=1}^n Z_{[i]} \leq -n\mu \leq a\sigma\sqrt{n} - \frac{1}{n}\sum_{i=1}^n Z_{[i]}\right) \\ &= P\left(\hat{\mu}_n - \frac{a\sigma}{\sqrt{n}} \leq \mu \leq \hat{\mu}_n + \frac{a\sigma}{\sqrt{n}}\right). \end{aligned}$$

Combining the above shows that if W is a standard normal and n is sufficient large[2] we have that

$$P\left(\hat{\mu}_n - \frac{a\sigma}{\sqrt{n}} \leq \mu \leq \hat{\mu}_n + \frac{a\sigma}{\sqrt{n}}\right) \approx P(-a \leq W \leq a).$$

For example, if $a = 1.96$, then

$$P(-1.96 \leq W \leq 1.96) \approx 0.95,$$

and so

$$\left[\hat{\mu}_n - \frac{1.96\sigma}{\sqrt{n}} \leq \mu \leq \hat{\mu}_n + \frac{1.96\sigma}{\sqrt{n}}\right]$$

is called the 95% confidence interval. Thus, we know that there is a 95% probability that the actual value is within our bounds (generated by our numerical method). Other sized confidence intervals can be constructed in the obvious manner.

Note that the confidence interval depends upon σ, the standard deviation of Z. Of course, if we don't know $\mathbb{E}Z$, we most likely do not know σ. Thus, one uses the estimated standard deviation

$$s_n = \sqrt{\frac{1}{n-1}\sum_{i=1}^n (Z_{[i]} - \mu_n)^2},$$

[2]We will not discuss how to know when "n is sufficiently large"

and the confidence interval becomes

$$\left[\hat{\mu}_n - \frac{as_n}{\sqrt{n}} \leq \mu \leq \hat{\mu}_n + \frac{as_n}{\sqrt{n}}\right].$$

Note, therefore, that our error, ϵ, scales like $1/\sqrt{n}$,

$$\epsilon = \frac{a}{s_n}\sqrt{n}.$$

This error is often called the *statistical error*. In the next section we will be introduced to a different form of error, the *bias*, which we will then study for the rest of these notes. Improvements on the statistical error can sometimes be achieved through quasi-Monte Carlo methods.

Biased estimators

The previous section assumed that $Z_{[i]}$ could be computed with the exact distribution. Oftentimes this is not possible and there is a *bias*. Letting $Z_{[i]}$ be the realizations, we define

$$\text{bias} = \mathbb{E}Z - \mathbb{E}Z_{[i]}.$$

Clearly, if the bias is non-zero then we no longer have

$$\hat{\mu}_n \to \mathbb{E}Z, \text{ as } n \to \infty.$$

However if the bias is small, this may not be too important. Now we must recognize that there will be two sources of error to any numerical study: one coming from the bias of the method, and the other coming from the statistical error. That is, we have

$$\mathbb{E}Z - \hat{\mu}_n = (\mathbb{E}Z - \mathbb{E}Z_{[i]}) + (\mathbb{E}Z_{[i]} - \hat{\mu}_n),$$

where we recognize the first error as the bias and the second as the statistical error. Now, for a given desired accuracy of $\epsilon > 0$, we would ideally like to ensure that both sources of error are below $\epsilon/2$.

Estimating solutions to SDEs

While it is sometimes possible to solve an SDE exactly, and hence be able to utilize an unbiased estimator, this is an exceedingly rare occurrence. However, we know it is possible by the following example.

Example 1 (Geometric Brownian Motion)**.** If X_t satisfies

$$dX_t = \mu X_t dt + \sigma X_t dB_t,$$

then we have already seen that

$$X_t = X_0 \exp\left\{\left(\mu - \frac{\sigma^2}{2}\right)t + \sigma B_t\right\}.$$

Hence, to approximate $\mathbb{E}f(X(t)$, for known function f and a fixed t, we could use

$$\hat{\mu}_n = \frac{1}{n}\sum_{i=1}^{n} X_0 \exp\left\{\left(\mu - \frac{\sigma^2}{2}\right)t + \sigma\sqrt{t}W_{[i]}\right\},$$

where $W_{[i]}$ are independent standard normal random variables (why?).

Note that we still can not produce an unbiased estimator to random variables of the form

$$\int_0^t f(X_s)ds,$$

since we can not observe X_s for all s. This will add a third type of error: discretization error. □

Of course, most SDEs will not have a nice solution.

Example 2. We already saw that an important SDE from finance is one of the form

$$dX_t = \kappa(\theta - X_t)dt + \sigma\sqrt{X_t}dB_t.$$

This SDE arose as a model for volatility in the Heston model and does not satisfy our Lipschitz conditions. However, it can be shown that there is a unique solution. □

Our First Method: Euler-Maruyama

We recall our one dimensional SDE

$$dX_t = a(t, X_t)dt + b(t, X_t)dB_t,$$

or

$$X_t = X_0 + \int_0^t a(s, X_s)ds + \int_0^t b(s, X_s)dB_s.$$

By far the simplest numerical scheme (and also by far the most widely used) to generate an approximate path of the above equation is Euler-Maruyama. First, we discretize $[0, T]$ into N equally sized sub-intervals

$$0 = t_0 < t_1 < t_2 < \cdots < t_{N-1} < t_N = T,$$

where

$$t_i = i\frac{T}{N}.$$

Let $h = T/N = t_n - t_{n-1}$. Then, we define $Z_0 = x_0$ and for $n \leq N$ set

$$Z_n = Z_{n-1} + a(t_{n-1}, Z_{n-1})h + b(t_{n-1}, Z_{n-1})\left(\sqrt{h}\cdot\eta_{[i]}\right),$$

where the $\eta_{[i]}$ are independent standard normal random variables. Note that we have used the relation

$$B_{t_i} - B_{t_{i-1}} \sim N(0, h) \sim \sqrt{h}N(0, 1),$$

and we are using the independence of the increments of a Brownian motion. Thus, we note that while Z_n is an approximation to X_{t_n}, we are simulating B_t, the driving Brownian motion, exactly at the time points t_n. Therefore, another way to write Euler-Maruyama is

$$Z_n = Z_{n-1} + a(t_{n-1}, Z_{n-1})\Delta t + b(t_{n-1}, Z_{n-1})\Delta_n B_t.$$

It is worth explicitly noting where this scheme comes from. Essentially, it comes from a first order Taylor expansion, at each time point, of the coefficients. That is, we have that

$$\begin{aligned} X(t_n) &= X(t_{n-1}) + \int_0^h a(s, X(s))ds + \int_0^h b(s, X(s))dB_s \\ &\approx X(t_{n-1}) + \int_0^h a(t_{n-1}, X(t_{n-1}))ds + \int_0^h b(t_{n-1}, X(t_{n-1}))dB_s \\ &= X(t_{n-1}) + a(t_{n-1}, X(t_{n-1}))h + b(t_{n-1}, X(t_{n-1}))\Delta_n B_t. \end{aligned}$$

Convergence

There are multiple types of convergence. Let Z^h denote the approximate process with a step size of $h = T/n > 0$. We say that the method converges **strongly** if

$$\lim_{h\to 0} \mathbb{E}|X(T) - Z^h(T)| = 0$$

We say it converges **weakly** if

$$\lim_{h\to 0} |\mathbb{E}f(X(T) - \mathbb{E}f(Z^h(T))| = 0,$$

for a large class of functions f (say, all differentiable functions with compact support).

Note that so far Z is only defined at the time discretization points. However, it is often useful to enlarge Z to a process defined on all of $[0, T]$ defined in the following manner. For $s \in [t_i, t_{i+1})$,

$$\begin{aligned} Z(s) &= Z(t_i) + \int_{t_i}^s a(t_i, Z(t_i))ds + \int_{t_i}^s b(t_i, Z(t_i))dB_s \\ &= Z(t_i) + a(t_i, Z(t_i))(s - t_i) + b(t_i, Z(t_i)) \cdot (B(s) - B(t_i)). \end{aligned}$$

A stronger form of strong convergence would then be to have

$$\lim_{h\to 0} \mathbb{E} \sup_{s\le T} |X(s) - Z^h(T)| = 0.$$

Of course, once we have a notion of convergence, the next natural question is: how fast does the method converge to the exact solution. We say that the method converges in the strong sense with an order of $p > 0$ if

$$\limsup_{h\to 0} \mathbb{E}|X(T) - Z^h(T)|h^{-p} \le C,$$

for some $C > 0$. This is equivalent to the existence of some $C > 0$ for which

$$\mathbb{E}|X(t) - Z^h(t)| \leq Ch^p,$$

for all $t \leq T$. Further, we need not consider only the L^1 norm. Thus, we say the method is strongly convergent in L^q with an order of $p > 0$ if there is a $C > 0$ for which

$$\left(\mathbb{E}|X(t) - Z^h(t)|\right)^{1/q} \leq Ch^p,$$

for all $t \leq T$, or if

$$\left(\mathbb{E}\sup_{s\leq T}|X(s) - Z^h(T)|^q\right)^{1/q} \leq Ch^p.$$

Of course, we say it converges weakly with an order of $p > 0$ if

$$|\mathbb{E}f(X(T)) - \mathbb{E}f(Z^h(T))| \leq Ch^p.$$

Convergence order for Euler's method

We start by defining the step function η via

$$\eta(s) = t_i, \quad \text{if } s \in [t_i, t_{i+1}).$$

We then have

$$X(t) - Z(t) = \int_0^t a(s, X(s)) - a(\eta(s), Z(\eta(s)))ds + \int_0^t b(s, X(s)) - b(s, Z(\eta(s)))dB(s).$$

Hence, using our Lipschitz assumption,

$$\mathbb{E}|X(t) - Z(t)|^2 \leq 2C\int_0^t \mathbb{E}|X(s) - Z(\eta(s))|^2 ds + 2C\int_0^t \mathbb{E}|X(s) - Z(\eta(s))|^2 ds,$$

where we used Itô's lemma. Hence, adding and subtracting $Z(s)$ from each term yields,

$$\mathbb{E}|X(t) - Z(t)|^2 \leq C_1\int_0^t \mathbb{E}|X(s) - Z(s)|^2 ds + C_2\int_0^t \mathbb{E}|Z(s) - Z(\eta(s))|^2 ds.$$

It is relatively straightforward to estimate $\mathbb{E}|Z(s) - Z(\eta(s))|^2$. Assuming $s \in [t_i, t_{i+1})$,

$$\mathbb{E}|Z(s) - Z(\eta(s))|^2 = \mathbb{E}\left(a(t_i, Z(t_i))(s - t_i) + b(t_i, Z(t_i)) \cdot (B(s) - B(t_i))\right)^2 = O(h),$$

where the leading order term came from the Brownian motion (note that if the process were deterministic, this value would have been h^2). Thus, collecting the above shows that for some $K_1, K_2 > 0$,

$$\mathbb{E}|X(t) - Z(t)|^2 \leq K_1 h + K_2\int_0^t \mathbb{E}|X(s) - Z(s)|^2 ds.$$

Gronwall's inequality now shows that

$$\mathbb{E}|X(t) - Z(t)|^2 \leq K_1 h e^{K_2 t} \implies \left(\mathbb{E}|X(t) - Z(t)|^2\right)^{1/2} \leq K_1^{1/2} h^{1/2} e^{K_2 t/2}$$

and the method converges at a rate of $O(h^{1/2})$.

It can also be shown that Euler's method converges in a weak sense at a rate of $O(h)$. This is a substantial improvement over the strong convergence rate.

Milstein Scheme

A large drawback of Euler's method is the low strong convergence rate. This can be alleviated by use of the Milstein scheme, which we present here in a simplified setting. We now limit ourselves to the following type of SDE

$$dX_t = a(X_t)dt + b(X_t)dB(t),$$

or

$$X_t = X_0 + \int_0^t a(X_s)ds + \int_0^t b(X_s)dB(s).$$

We now search for a higher order scheme by applying Itô to the coefficients and keeping only the higher order terms. Thus, for any discretization we have

$$\begin{aligned}
X_{t_{i+1}} &= X_{t_i} + \int_{t_i}^{t_{i+1}} \left[a(X_{t_i}) + \int_{t_i}^{s} a'(X_r)dX_r + \frac{1}{2}\int_{t_i}^{s} a''(X_r)(dX_r)^2 \right] ds \\
&\quad + \int_{t_i}^{t_{i+1}} \left[b(X_{t_i}) + \int_{t_i}^{s} b'(X_r)dX_r + \frac{1}{2}\int_{t_i}^{s} b''(X_r)(dX_r)^2 \right] dB_s \\
&= X_{t_i} + \int_{t_i}^{t_{i+1}} \left[a(X_{t_i}) + \int_{t_i}^{s} a'(X_r)a(X_r)dr + \int_{t_i}^{s} a'(X_r)b(X_r)dB_r \right] ds \\
&\quad + \frac{1}{2}\int_{t_i}^{t_{i+1}} \int_{t_i}^{s} a''(X_r)b(X_r)^2 dr ds \\
&\quad + \int_{t_i}^{t_{i+1}} \left[b(X_{t_i}) + \int_{t_i}^{s} b'(X_r)a(X_r)dr + \int_{t_i}^{s} b'(X_r)b(X_r)dB(r) \right] dB(s) \\
&\quad + \frac{1}{2}\int_{t_i}^{t_{i+1}} \int_{t_i}^{s} b''(X_r)b(X_r) dr dB_s.
\end{aligned}$$

Keeping only terms of the form ds and $dB_s dB_r$ (which should be $O(h)$), we have

$$\begin{aligned}
X_{t_{i+1}} &= X_{t_i} + \int_{t_i}^{t_{i+1}} a(X_{t_i})ds + \int_{t_i}^{t_{i+1}} b(X_{t_i})dB(s) + \int_{t_i}^{t_{i+1}} \int_{t_i}^{s} b'(X_r)b(X_r)dB(r)dB_s \\
&= X_{t_i} + a(X_{t_i})h + b(X_{t_i})\Delta_i B + \int_{t_i}^{t_{i+1}} \int_{t_i}^{s} b'(X_r)b(X_r)dB(r)dB_s.
\end{aligned}$$

The first terms are simply Euler's method. The next term can be approximate further (think about taking another Itô expansion) by

$$\begin{aligned}
\int_{t_i}^{t_{i+1}} \int_{t_i}^{s} b'(X_r)b(X_r)dB(r)dB(s) &\approx b'(X(t_i))b(X(t_i)) \int_{t_i}^{t_{i+1}} \int_{t_i}^{s} dB(r)dB(s) \\
&= b'(X(t_i))b(X(t_i)) \int_{t_i}^{t_{i+1}} \int_{t_i}^{s} (B(s) - B(t_i))dB(s) \\
&= b'(X(t_i))b(X(t_i)) \left(\frac{1}{2}(B(t_{i+1}) - B(t_i))^2 - \frac{1}{2}h \right) \\
&= \frac{1}{2} b'(X(t_i))b(X(t_i)) \left((\Delta_i B)^2 - h \right).
\end{aligned}$$

Hence, the Milstein scheme is

$$Z_n = Z_{n-1} + a(Z_{n-1})h + b(Z_{n-1})(\Delta_{n-1}B) + \frac{1}{2}b'(Z(t_n))b(Z(t_n))((\Delta_{n-1}B)^2 - h).$$

The Milstein scheme has a strong convergence rate of $O(h)$ and a weak rate of $O(h)$.

Computational complexity

We are now ready to consider the total amount of work, or CPU time required to solve a given problem. We will only consider the problem in terms of "order of magnitude" and not worry about constants. That is, if we say that the total work is of order ϵ^{-2} for some small $\epsilon > 0$, then we mean there is a constant for which the total work is below $C\epsilon^{-2}$.

Let's again consider the problem of approximating

$$\mathbb{E}f(X(T)),$$

and we want to be accurate, in the sense of confidence intervals, to an order of $\epsilon > 0$. Let Z_h denote a path generated by an approximate algorithm with a time discretization parameter of $h > 0$. Let

$$\hat{\mu}_n = \frac{1}{n}\sum_{i=1}^{n} f(Z_{[i]}(T)).$$

As before, we have

$$\mathbb{E}f(X(T)) - \hat{\mu}_n = (\mathbb{E}f(X(T)) - \mathbb{E}f(Z(T))) + (\mathbb{E}f(Z(T)) - \hat{\mu}_n),$$

with the first term called the *bias* of the method, and the second is the statistical error. Let's suppose that the scheme being used is accurate, in a weak sense, of $O(h)$. That is,

$$|\mathbb{E}f(X(T)) - \mathbb{E}f(Z(T))| \leq Ch.$$

Then, to get the bias to be $O(\epsilon)$, we need to choose $h = O(\epsilon)$. How many paths do we need to generate? We need the standard deviation of the estimator, which gives the width of the confidence interval, to be $O(\epsilon)$. Hence, we need the variance to be $O(\epsilon^2)$. The variance is

$$Var(\hat{\mu}_n) = \frac{1}{n}Var(f(Z_i(T))).$$

Thus, we require

$$\frac{1}{n} = O(\epsilon^2) \implies n = \epsilon^{-2}.$$

Thus, we need $O(\epsilon^{-2})$ paths, each with a step size of $O(\epsilon)$. The work for each path will be some constant time $1/h$ (number of steps), thus the total amount of work, or computational complexitiy (and hence CPU time) is

$$O(\epsilon^{-3}).$$

What if we had had a higher order method (and they do exist). What if the *weak error* of the method were $O(h^2)$? Then,

1. We still need $O(\epsilon^{-2})$ paths, to control the statistical error.
2. For the bias, we need $h^2 = \epsilon \implies h = O(\sqrt{\epsilon})$. Hence, the work per path is now only $\epsilon^{-1/2}$..

Thus, the total computational complexity is now

$$\epsilon^{-2}\epsilon^{-1/2} = \epsilon^{-2.5}.$$

On the other hand, consider the situation in which the numerical accuracy where only $h^{1/2}$. For example, you may be solving a problem like

$$\mathbb{E}\int_0^t f(X(s))ds,$$

which requires information of the whole path. Then Euler is only $O(h^{1/2})$ accurate. In this case we need to require that $h^{1/2} = \epsilon \implies h = \epsilon^2$. Now the computational complexity would be

$$O(\#\text{paths}) \times O(\text{cost per path}) = O(\epsilon^{-2}\epsilon^{-2}) = O(\epsilon^{-4}),$$

which is terrible for small ϵ. Here, Milstein would be beneficial.

Multilevel Monte Carlo

The following is a novel method for the computation of expectations. In its simplest version, we suppose we want to solve

$$\mathbb{E}f(X(T)),$$

to an accuracy of $\epsilon > 0$. We suppose that Z_h is an Euler approximation, and thus has a bias of $O(h)$.

The idea is exceedingly simple, and is essentially a *control variate* approach. For some $L > 0$ we let

$$h_L = \frac{1}{2^L}.$$

We choose L so that $h_L = \epsilon$. Note, therefore, that $L \approx \frac{1}{\ln(2)}|\ln(\epsilon)|$. Thus, we have already controlled the bias. However, the statistical error is still a problem.

Next, we let

$$h_\ell = \frac{1}{2^\ell}.$$

Then, we just make the very simple observation that

$$\begin{aligned}
\mathbb{E}f(Z_L) &= [\mathbb{E}f(Z_L(T)) - f(Z_{L-1}(T))] + \mathbb{E}f(Z_{L-1}(T)) \\
&= [\mathbb{E}f(Z_L(T)) - f(Z_{L-1}(T))] + [\mathbb{E}f(Z_{L-1}(T)) - \mathbb{E}f(Z_{L-2}(T))] + \mathbb{E}f(Z_{L-2}(T)) \\
&\vdots \\
&= \mathbb{E}f(Z_0(T)) + \sum_{\ell=1}^{L} \left[\mathbb{E}f(Z_\ell(T)) - f(Z_{\ell-1}(T))\right].
\end{aligned}$$

Now, you construct Z_ℓ and $Z_{\ell-1}$ simultaneously (using same Brownian increments), to *couple* them. The variances will be small. (We won't prove this here.) In fact, the computational complexity is reduced to

$$O(\epsilon^{-2}\log(|\epsilon|)^2),$$

which is nearly as low as $O(\epsilon^{-2})$, which is computational complexity if we could simply sample directly from $X(T)$.