

NUMERICAL METHODS FOR SDE SIMULATION

by SILVIA LAVAGNINI

BI Norwegian Business School

14 May 2024

SDE

In fact, the asset price $S(t)$ follows a GBM process when it satisfies the following SDE:

$$\begin{cases} dS(t) = \mu S(t) dt + \sigma S(t) dB(t) \\ S(t_0) = S_0 \end{cases}$$

More in general, we can consider a so-called $\hat{\text{It\^o}}$ process:

Definition 2.1.3 (Itô process) Let us consider the following SDE, corresponding to the Itô process $X(t)$,

$$dX(t) = \bar{\mu}(t, X(t))dt + \bar{\sigma}(t, X(t))dW(t), \quad \text{with } X(t_0) = X_0, \quad (2.5)$$

with two general functions for the drift $\bar{\mu}(t, x)$ and the volatility $\bar{\sigma}(t, x)$. These two functions cannot be “just any” functions; they need to satisfy the following

Lipschitz conditions:

and growth

$$|\bar{\mu}(t, x) - \bar{\mu}(t, y)|^2 + |\bar{\sigma}(t, x) - \bar{\sigma}(t, y)|^2 \leq K_1 |x - y|^2,$$

$$|\bar{\mu}(t, x)|^2 + |\bar{\sigma}(t, x)|^2 \leq K_2(1 + |x|^2),$$

for some constants $K_1, K_2 \in \mathbb{R}^+$ and x and y in \mathbb{R} . The two conditions above state that the drift and volatility terms should not increase too rapidly. When these conditions hold, then, with probability one, a continuous, adapted solution of (2.5) exists, and the solution satisfies $\sup_{0 \leq t \leq T} \mathbb{E}[X^2(t)] < \infty$. \blacktriangleleft

The solution of (2.5) is given by

$$X(T) = X_0 + \int_{t_0}^T \bar{\mu}(t, X(t)) dt + \int_{t_0}^T \bar{\sigma}(t, X(t)) dW(t). \quad (9.12)$$

Generally, when both sides of (9.12) contain the term $X(t)$, the SDE needs to be solved numerically. Only for a few standard processes, an explicit solution is available.

Goal construct and analyze numerical methods for SDEs.

For this purpose, we define a partition of the integration interval : choose $m \in \mathbb{N}$ and define the step size

$$\Delta t := \frac{T}{m}, \text{ so that we obtain the partition}$$

$$0 = t_0 < t_1 < \dots < t_m = T \quad \text{where } t_i = i \Delta t.$$

Then, for each time points t_i we obtain :

$$x_{i+1} = x_i + \int_{t_i}^{t_{i+1}} \bar{\mu}(t, X(t)) dt + \int_{t_{i-1}}^{t_i} \bar{\sigma}(t, X(t)) dW(t), \quad (9.13)$$

where $x_i = X(t_i)$. Numerical integration schemes can be employed to approximate the integrals in (9.13). ~~The order of convergence of an integration scheme is defined as~~

Stochastic Euler scheme (Euler - Maruyama)

We approximate (9.13) by approximating the integrand at its initial value:

$$\begin{aligned}x_{i+1} &\approx x_i + \bar{\mu}(t_i, x_i)(t_{i+1} - t_i) + \bar{\sigma}(t_i, x_i)(W(t_{i+1}) - W(t_i)) \\&\stackrel{d}{=} x_i + \bar{\mu}(t_i, x_i)\Delta t + \bar{\sigma}(t_i, x_i)W(\Delta t),\end{aligned}\quad (9.14)$$

where $x_i \approx X(t_i)$ and $x_0 = X(t_0)$.

In particular, the quantity $W(\Delta t) \sim N(0, \Delta t)$, hence

$$x_{i+1} \approx x_i + \bar{\mu}(t_i, x_i)\Delta t + \bar{\sigma}(t_i, x_i)\sqrt{\Delta t}Z.\quad (9.15)$$

for $Z \sim N(0, 1)$.

Notice that Euler approximation is a continuous-time stochastic process satisfying the iterative scheme (9.14).

In particular, the sequence $\{x_n, n=0, \dots, m\}$ represents the approximate values of the Itô process at the discretization times only. If required, values can then be determined at the intermediate instants by an appropriate interpolation method. Example:

- Piecewise constant interpolation :

$$x(t) = x_{n_t} \quad \text{where} \quad n_t := \max \{ n=0, 1, \dots, m : t_n \leq t \}$$

- Linear interpolation :

$$x(t) = x_{n_t} + \frac{t - t_{n_t}}{t_{n_{t+1}} - t_{n_t}} (x_{n_{t+1}} - x_{n_t}) .$$

Remark: for $\bar{\mu}(t, x) = 0$, the Euler-Maruyama method reduces to the explicit Euler method applied to ODEs.

Notice that the exact solution $X(t_m)$ and the approximation x_m are random variables, hence, for every path $t \mapsto B(t, w)$ of the BM, a different result is obtained.

EXAMPLE: Euler scheme for GBM

The GBM process with dynamics, $dS(t) = \mu S(t)dt + \sigma S(t)dW(t)$, has an exact solution in the time interval $[t_i, t_{i+1}]$,

$$S(t_{i+1}) = S(t_i) \exp \left((\mu - \frac{1}{2}\sigma^2)\Delta t + \sigma (W(t_{i+1}) - W(t_i)) \right). \quad (9.17)$$

The Euler discretization for the i -th time step is

$$(9.16) \quad \begin{cases} S_{i+1} = S_i + \mu S_i \cdot \Delta t + \sigma S_i (W_{i+1} - W_i), & i = 0, \dots, m-1 \\ S_0 = S(t_0) \end{cases}$$

CODE

(The exact solution should use $m=2^9$ discretization points.)

We consider the Euler scheme with $m=2^2$ points;

there $m=2^4$ points. We note that at $T=1$ the Euler approximation is closer than for the first experiment.

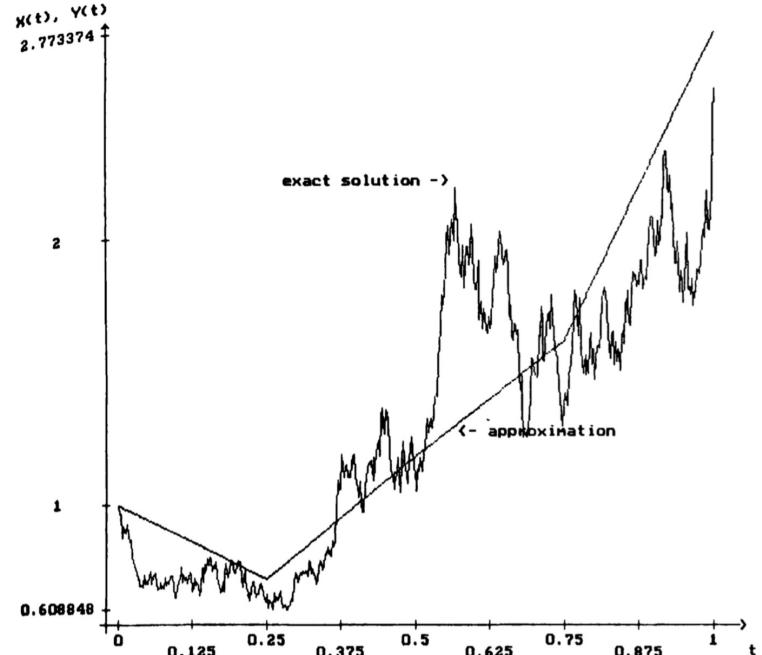


Figure 9.2.1 Euler approximation and exact solution from PC-Exercise 9.2.1.

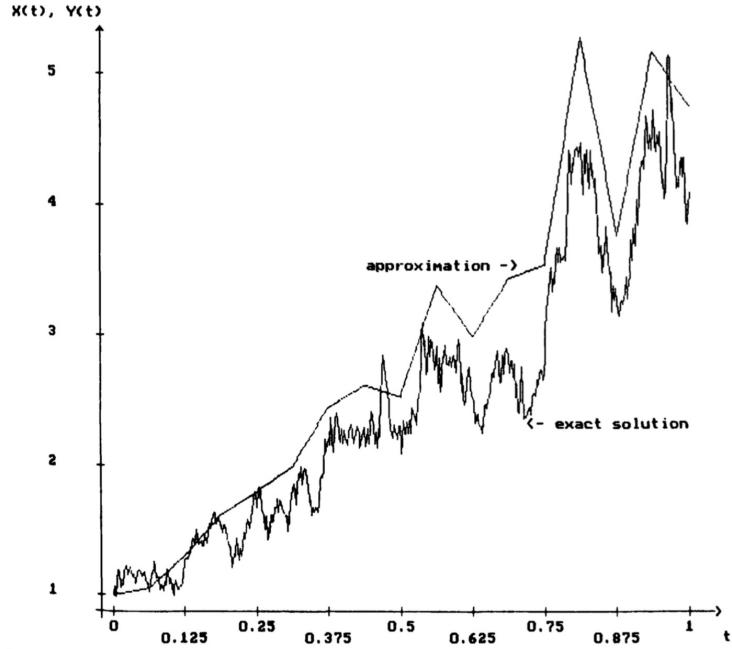


Figure 9.2.2 The Euler approximation for the smaller step size $\Delta = 2^{-4}$.

We need to define a criterion to judge the quality / accuracy of the approximation. Such a criterion must reflects the main goal of a practical simulation.

There are two basic tasks connected to simulations of solutions of SDEs:

- The first occurs in situations where a good pathwise approximation is required (direct simulations, filtering problems, testing statistical estimators, ...)
- The second focuses on approximating expectations of functionals of the I^{to}s process, e.g. its distribution or its moments. This is relevant when such functionals can not be determined analytically.

Pathwise approximation

In case we know the explicit solution of our SDE, we can compute the approximation error using the ABSOLUTE ERROR CRITERION

$$\varepsilon = \mathbb{E}[|X(T) - x^m|] \quad (3.1)$$

which gives a measure of the pathwise closeness at the end of the time interval $[0, T]$.

We examine the absolute error criterion numerically.

We repeat N different simulations denoted by $s_{m,j}$ for the Euler scheme and $s_j^*(T)$ for the exact one, $j = 1, \dots, N$. (here $s_{i,j}$ is the Euler discretization for the i -th timestep and j -th path, $i = 0, \dots, m$ and $j = 1, \dots, N$).

For different Δt -values, the absolute error criterion is estimated by:

$$\hat{\mathcal{E}}(\Delta t) = \frac{1}{N} \sum_{j=1}^N |S_j(T) - s_{m,j}| = \frac{1}{N} \sum_{j=1}^N |S(t_0)e^{(\mu - \frac{1}{2}\sigma^2)T + \sigma W_{m,j}} - s_{m,j}|,$$

CODE

Simulate $N=25$ trajectories for the Itô process (9.17) and their corresponding Euler approximations with time steps: $\Delta t = 2^{-4}, 2^{-5}, 2^{-6}, 2^{-7}$.

Repeat the experiment with several seeds.

Note: for this experiment, it is crucial to use the same BM for both the exact and approximated solution.

Δ	2^{-4}	2^{-5}	2^{-6}	2^{-7}
$\hat{\epsilon}_1$	0.5093	0.4446	0.3265	0.2292
$\hat{\epsilon}_2$	0.4692	0.3788	0.2234	0.1477

Table 9.3.2 Absolute errors $\hat{\epsilon}_1$ and $\hat{\epsilon}_2$ for different step lengths Δ .

COMMENTS :

- the estimate of the absolute error decreases with decreasing time step ;
- However, these estimates are random variables.
- By the central limit theorem, for large N , the error $\hat{\epsilon}$ becomes asymptotically a Gaussian r.v. and converges in distribution to the non-random value ϵ as $N \rightarrow +\infty$.
- We can not simulate infinite trajectories, but we can estimate the variance $\sigma_{\hat{\epsilon}}^2$ of $\hat{\epsilon}$, and use it to construct a confidence interval for ϵ .

How: •) We arrange the simulations into M batches of N simulations each.

••) For each batch we compute

$$\hat{\varepsilon}_k = \frac{1}{N} \sum_{j=1}^N |S_{j,k}(T) - s_{m,j,k}|, \quad k=1, \dots, M.$$

The $\hat{\varepsilon}_k$ are independent and approximately Gaussian for large N .

•••) We estimate mean and variance

$$\hat{\varepsilon} = \frac{1}{M} \sum_{k=1}^M \hat{\varepsilon}_k = \frac{1}{NM} \sum_{k=1}^M \sum_{j=1}^N |S_{j,k}(T) - s_{m,j,k}| \quad (3.4)$$

$$\hat{\sigma}_{\hat{\varepsilon}}^2 = \frac{1}{M-1} \sum_{j=1}^M (\hat{\varepsilon}_j - \bar{\hat{\varepsilon}})^2.$$

Since the $\hat{\varepsilon}_k$ are approximately Gaussian, we can use the Student t-distribution to construct confidence intervals for a sum of independent Gaussian (usually, the batch averages can be considered Gaussian for $N \geq 15$). We obtain the confidence interval

$$(\bar{\hat{\varepsilon}} - \Delta \hat{\varepsilon}, \bar{\hat{\varepsilon}} + \Delta \hat{\varepsilon})$$

with $\Delta \hat{\varepsilon} = t_{1-\alpha/2, M-1} \sqrt{\frac{\hat{\sigma}_{\hat{\varepsilon}}^2}{M}}$

$\underbrace{\quad}_{\hookrightarrow}$ t-distribution with $M-1$ degrees of freedom.

CODE

1) We simulate $M=10, 20, 40, 100$ batches with $N=100$ trajectories with step size $\Delta = 2^{-4}$, and determine the 95% C.I. against M .

↳ the length of the CI decreases as M increases.

2) Simulate $M=20$ batches with $N=100$ trajectories, using $\Delta t = 2^{-2}, 2^{-3}, 2^{-4}, 2^{-5}$, and plot the CI against Δt .

↳ we obtain $\tilde{\epsilon}(\Delta t) = K \Delta t^{1/2}$ for appropriate K .

We can also plot \log_2 versus \log_2 coordinates, for which the plot becomes a straight line with slope $\frac{1}{2}$.

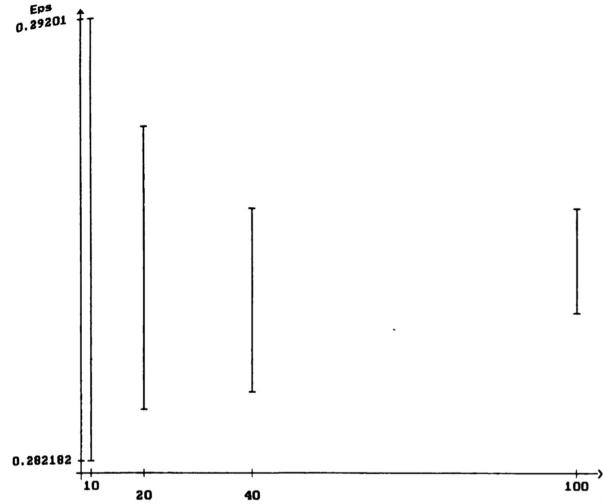


Figure 9.3.1 Confidence intervals for increasing numbers of batches.

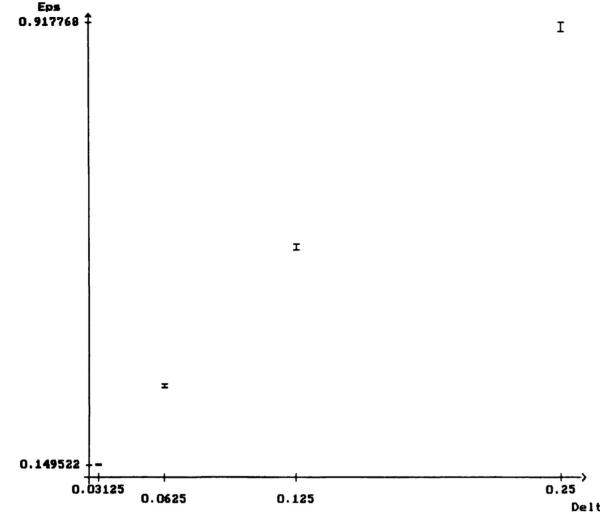


Figure 9.3.2 Confidence intervals for increasing step size.

\Rightarrow For a general Itô process,
Euler has an order of strong
convergence $1/2$.

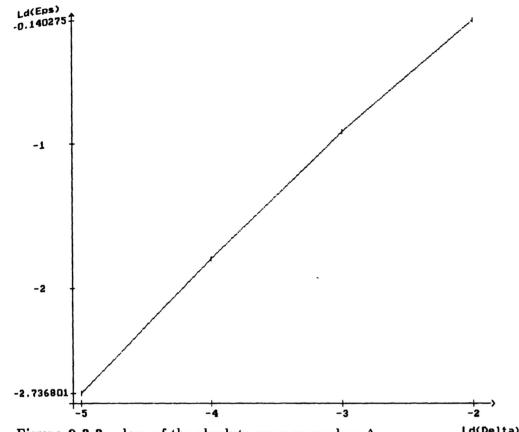


Figure 9.3.3 \log_2 of the absolute error versus $\log_2 \Delta$.

Ld(Delta)

Definition 9.1.1 (Convergence) Denote by x_m the approximation for $X(T)$, where Δt is the time step size, and m corresponds to the last term in the time discretization, $t_i = i \cdot \Delta t$, $i = 0, \dots, m$. Then, the approximation x_m converges in a strong sense to $X(T)$, with order $\alpha > 0$, if $\exists C > 0$ independent of Δt s.t.

$$\epsilon^s(\Delta t) := \mathbb{E}^\bullet[|x_m - X(T)|] \in O(h^\alpha) \leq C \Delta t^\alpha \text{ for all sufficiently small } \Delta t.$$

For a sufficiently smooth function $g(\cdot)$, the approximation x_m converges in a weak sense to $X(T)$, with respect to $g(\cdot)$, with order $\beta > 0$, if $\exists C > 0$ independent of Δt s.t.

$$\epsilon^w(\Delta t) := |\mathbb{E}^\bullet[g(x_m)] - \mathbb{E}^\mathbb{Q}[g(X(T))]| \in O(h^\beta) \leq C \Delta t^\beta \text{ for all sufficiently small } \Delta t.$$

In other words, a numerical integration method converges in a strong sense, if the asset prices converge, and weak convergence implies a convergent approximation of the probability distribution of $X(T)$, for a given time T . The convergence then concerns only the marginal distribution of $X(T)$. ◀

We can decompose the r.v. $\hat{\varepsilon}$ (mean of batch averages) in two parts:

$$\hat{\varepsilon} = \underbrace{\varepsilon_{\text{sys}}}_{\parallel} + \underbrace{\varepsilon_{\text{stat}}}_{\text{Statistical error}} \Rightarrow \text{Statistical error} = \hat{\varepsilon} - \mathbb{E}[\hat{\varepsilon}]$$

$\mathbb{E}[\hat{\varepsilon}]$ is the systematic error

- From (3.1) and (3.4)

$$\begin{aligned}\varepsilon_{\text{sys}} &= \mathbb{E}[\hat{\varepsilon}] = \mathbb{E}\left[\frac{1}{NM} \sum_{k=1}^M \sum_{j=1}^N |S_{j,k}(T) - S_{m,j,k}| \right] \\ &= \mathbb{E}[|S(T) - S_m|] = \varepsilon \quad \parallel \text{the systematic error} \\ &\quad \text{coincides with the absolute error.}\end{aligned}$$

- From above, $\varepsilon_{\text{stat}} = \hat{\varepsilon} - \varepsilon$.

For a large number MN of independent simulations, the Central Limit theorem says that $\varepsilon_{\text{stat}}$ is asymptotically

Gaussian with mean zero and variance

$$\begin{aligned}\text{Var}(\hat{\epsilon}_{\text{stat}}) &= \text{Var}(\hat{\epsilon} - \epsilon) = \mathbb{E}[(\hat{\epsilon} - \epsilon)^2] \\&= \mathbb{E}\left[\left(\frac{1}{NM} \sum_{k=1}^M \sum_{j=1}^N |S_{j,k}(T) - s_{m,j,k}| - \epsilon\right)^2\right] \\&= \frac{1}{(NM)^2} \sum_{k=1}^M \sum_{j=1}^N \mathbb{E}\left[\left(|S_{j,k}(T) - s_{m,j,k}| - \epsilon\right)^2\right] \\&= \frac{1}{NM} \mathbb{E}\left[\left(|S(T) - s_m| - \epsilon\right)^2\right] = \frac{1}{NM} \text{Var}(|S(T) - s_m|) \\&\quad \epsilon = \mathbb{E}[|S(T) - s_m|]\end{aligned}$$

→ The statistical error depends on the total number NM .

Weak convergence

Let

$\mu = \mathbb{E}[S_m] - \mathbb{E}[S(T)]$ be the mean/weak error.

Notice that to estimate μ we do not need to use the same sample paths of the Wiener process when generating S and S_m . We require that the probability distributions of $S(T)$ and S_m are sufficiently closed to each other.

Let $s_{i,j}$ the Euler discretization for the i -th timestep and j -th path, $i = 0, \dots, m-1$ and $j = 1, \dots, N$.

Then, for different Δt -values, the weak convergence error at the maturity time T is

$$\begin{aligned}\hat{\mu}(\Delta t) &= \left| \frac{1}{N} \sum_{j=1}^N S_j(T) - \frac{1}{N} \sum_{j=1}^N s_{m,j} \right| \\ &= \left| S(t_0) \frac{1}{N} \sum_{j=1}^N e^{(r - \frac{1}{2}\sigma^2)T + \sigma W_{m,j}} - \frac{1}{N} \sum_{j=1}^N s_{m,j} \right|,\end{aligned}$$

as before, $\hat{\mu}$ is a random variable and we work with several batches.

For M batches of N trajectories, we estimate the mean error of the k -th batch by

$$\hat{\mu}_k = \frac{1}{N} \sum_{j=1}^N s_{m,j,k} - \mathbb{E}[S(T)] , \quad k=1, \dots, M$$

We could also compute analytically
 (where here $\mathbb{E}[\delta(t)] = \mathbf{x}_0 e^{\mu^T}$) and their average

$$\hat{\mu} = \frac{1}{M} \sum_{k=1}^M \hat{\mu}_k = \frac{1}{MN} \sum_{k=1}^M \sum_{j=1}^N s_{m,j,k} - \mathbb{E}[\delta(t)]$$

and their variance $\hat{\sigma}_{\mu}^2 = \frac{1}{M-1} \sum_{k=1}^M (\hat{\mu}_k - \hat{\mu})^2$.

We then obtain the confidence interval $(\hat{\mu} - \Delta\hat{\mu}, \hat{\mu} + \Delta\hat{\mu})$
 where $\Delta\hat{\mu} = t_{1-\alpha, M-1} \sqrt{\frac{\hat{\sigma}_{\mu}^2}{M}}$.

CODE

Generate $M=20$ batches of $N=100$ trajectories and
 determine the 90% confidence interval for μ using
 $\Delta t = 2^{-3}, 2^{-4}, 2^{-5}$. Plot the error against the length Δt .

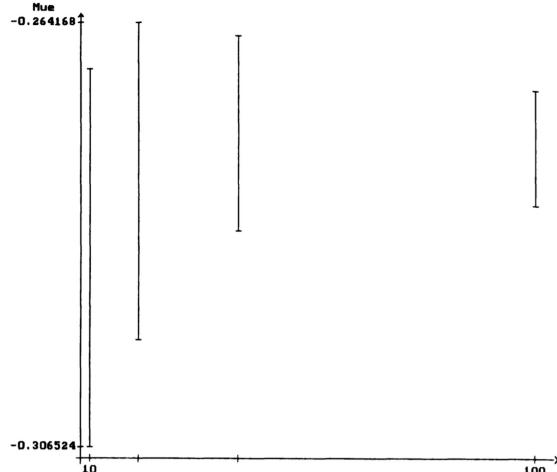


Figure 9.4.1 Confidence intervals for increasing number of batches.

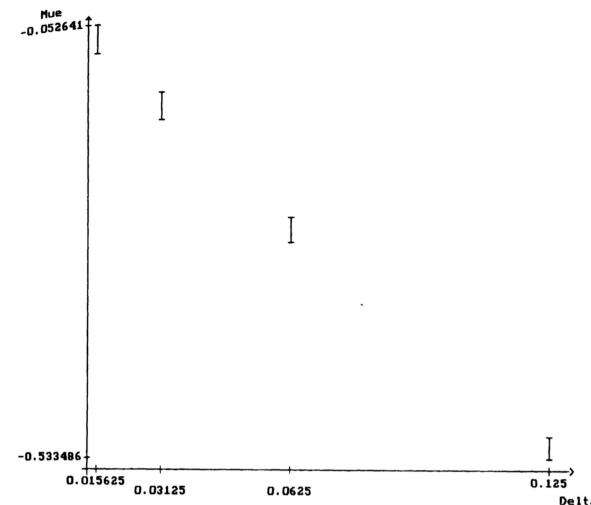


Figure 9.4.2 Confidence intervals for increasing step size.

↳ We could conclude from the graph that $\tilde{\mu}(\Delta t) \approx K \Delta t$ for an appropriate K . We can also plot the results in \log_2 - \log_2 coordinates. We obtain a straight line of slope 1.

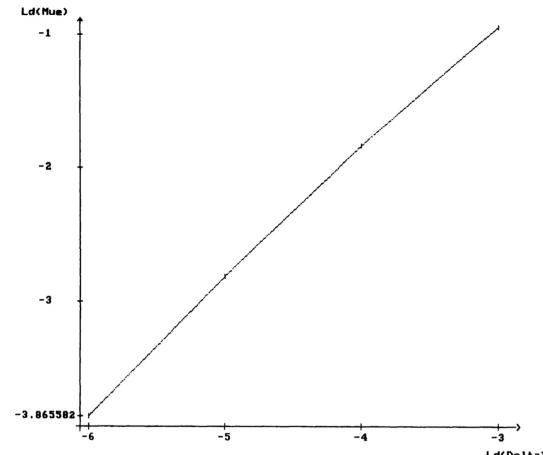


Figure 9.4.3 \log_2 of the absolute mean error versus $\log_2 \Delta$.

Similarly as before, we can decompose the random estimate $\hat{\mu}$ for the mean error μ into $\hat{\mu} = \mu_{\text{sys}} + \mu_{\text{stat}}$, where

$$\mu_{\text{sys}} = \mathbb{E}[\hat{\mu}] = \mathbb{E}\left(\frac{1}{MN} \sum_{k=1}^M \sum_{j=1}^N s_{m,j,k}\right) - \mathbb{E}[s_{CT}] = \mathbb{E}[s_m] - \mathbb{E}[s_{CT}] = \mu$$

and μ_{stat} , having, for large NM , asymptotically Gaussian distribution with mean zero and variance $\text{Var}(\varepsilon_{\text{stat}}) = \text{Var}(\hat{\mu}) = \frac{1}{MN} \text{Var}(s_m)$.

CONCLUSION : For the Euler scheme we have :

- strong convergence of order $1/2$;
- weak convergence of order 1 .

These holds under Lipschitz and linear growth conditions on the SDE coefficients.

Milstein scheme

For ODEs, one may employ the Taylor expansion to define discretizations by which we may obtain an higher order of convergence. For SDEs, we can do something similar by using the Itô-Taylor expansion.

For the Itô process SDE, $dX(t) = \bar{\mu}(t, X(t))dt + \bar{\sigma}(t, X(t))dW(t)$, the discretization under the *Milstein scheme* is obtained by adding a *third term* to the Euler discretization, i.e.,

$$\begin{aligned}x_{i+1} &= x_i + \int_{t_i}^{t_{i+1}} \bar{\mu}(t, X(t))dt + \int_{t_i}^{t_{i+1}} \bar{\sigma}(t, X(t))dW(t) \\&\approx x_i + \int_{t_i}^{t_{i+1}} \bar{\mu}(t_i, x_i)dt + \int_{t_i}^{t_{i+1}} \bar{\sigma}(t_i, x_i)dW(t) \\&\quad + \frac{1}{2}\bar{\sigma}(t_i, x_i)(W^2(\Delta t) - \Delta t)\frac{\partial \bar{\sigma}}{\partial x}(t_i, x_i),\end{aligned}$$

with $x_0 = X(t_0)$.

DERIVATION

Let $\bar{\mu}(t) := \bar{\mu}(t, X(t))$ and $\bar{F}(t) := F(t, X(t))$.

Since both $\bar{\mu}(t)$ and $\bar{F}(t)$ depend on $X(t)$, they are themselves stochastic variables and we can therefore derive their dynamics:

$$(9.18) \quad d\bar{\mu}(t) = \frac{\partial \bar{\mu}(t)}{\partial t} dt + \frac{\partial \bar{\mu}(t)}{\partial X} dX(t) + \frac{1}{2} \frac{\partial^2 \bar{\mu}(t)}{\partial X^2} (dX(t))^2 .$$

For $t \in [t_i, t]$, we integrate the processes (9.18) in the interval $[t_i, t]$:

$$\bar{\mu}(t) = \bar{\mu}(t_i) + \int_{t_i}^t \left(\underbrace{\frac{\partial \bar{\mu}}{\partial z}(z) + \bar{\mu}(z) \frac{\partial \bar{\mu}}{\partial X} + \frac{1}{2} \bar{F}^2(z) \frac{\partial^2 \bar{\mu}}{\partial X^2}}_{=: A_n(z)} \right) dz$$

$\therefore A_n(z)$

$$+ \int_{t_i}^t \bar{F}(z) \frac{\partial \bar{\mu}(z)}{\partial X} dB(z)$$

and similarly :

$$\bar{\sigma}(t) = \bar{\sigma}(t_i) + \int_{t_i}^t \left(\underbrace{\frac{\partial \bar{\sigma}}{\partial z}(z) + \bar{\mu}(z) \frac{\partial \bar{\sigma}}{\partial X}(z) + \frac{1}{2} \bar{\sigma}^2(z) \frac{\partial^2 \bar{\sigma}}{\partial X^2}(z)}_{=: A_2(z)} \right) dz$$
$$+ \int_{t_i}^t \bar{\sigma}(z) \frac{\partial \bar{\sigma}}{\partial X}(z) dB(z)$$

We substitute these last two into (9.13) and get :

$$x_{i+1} = x_i + \int_{t_i}^{t_{i+1}} \left[\bar{\mu}(t_i) + \int_{t_i}^t A_1(z) dz + \int_{t_i}^t \bar{\sigma}(z) \frac{\partial \bar{\mu}(z)}{\partial X} dW(z) \right] dt$$
$$+ \int_{t_i}^{t_{i+1}} \left[\bar{\sigma}(t_i) + \int_{t_i}^t A_2(z) dz + \int_{t_i}^t \bar{\sigma}(z) \frac{\partial \bar{\sigma}(z)}{\partial X} dW(z) \right] dW(t).$$

Neglecting all terms of higher order, i.e. $(dt)^2 = 0$ and $dt \cdot dW(t) = 0$, as in Table 2.1 in Chapter 2, the dynamics simplify to

$$\begin{aligned} x_{i+1} &= x_i + \int_{t_i}^{t_{i+1}} \bar{\mu}(t_i) dt + \int_{t_i}^{t_{i+1}} \bar{\sigma}(t_i) dW(t) \\ &\quad + \int_{t_i}^{t_{i+1}} \left(\int_{t_i}^t \bar{\sigma}(z) \frac{\partial \bar{\sigma}(z)}{\partial X} dW(z) \right) dW(t). \end{aligned} \quad (9.20)$$

The first two integrals in the expression above can be simplified, i.e.,

$$\begin{aligned} x_{i+1} &= x_i + \bar{\mu}(t_i) \Delta t + \bar{\sigma}(t_i) (W(t_{i+1}) - W(t_i)) \\ &\quad + \int_{t_i}^{t_{i+1}} \left(\int_{t_i}^t \bar{\sigma}(z) \frac{\partial \bar{\sigma}(z)}{\partial X} dW(z) \right) dW(t). \end{aligned} \quad (9.21)$$

For the inner integral in the last term, we apply the Euler discretization, as follows

$$\int_{t_i}^t \bar{\sigma}(z) \frac{\partial \bar{\sigma}(z)}{\partial X} dW(z) \approx \bar{\sigma}(t_i) \frac{\partial \bar{\sigma}}{\partial X}(t_i) (W(t) - W(t_i)).$$

As the expression under the outer integral in (9.21) is now constant in the interval $[t_i, t_{i+1}]$, it is placed outside the integral, giving

$$\bar{\sigma}(t_i) \frac{\partial \bar{\sigma}}{\partial X}(t_i) \int_{t_i}^{t_{i+1}} (W(t) - W(t_i)) dW(t). \quad (9.22)$$

Now, using that $\int_0^T W(t) dW(t) = \frac{1}{2} W^2(T) - \frac{1}{2} T$

we get (with some algebra)

$$\int_{t_i}^{t_{i+1}} (W(t) - W(t_i)) dW(t) = \frac{1}{2} (W(t_{i+1}) - W(t_i))^2 - \frac{1}{2} \Delta t.$$

After substitution, Equation (9.22) reads,

$$\int_{t_i}^{t_{i+1}} \left(\int_{t_i}^{t_{i+1}} \bar{\sigma}(z) \frac{\partial \bar{\sigma}(z)}{\partial X} dW(z) \right) dW(t) \approx \frac{1}{2} \bar{\sigma}(t_i) \frac{\partial \bar{\sigma}}{\partial X}(t_i) ((W(\Delta t))^2 - \Delta t).$$

The following discrete dynamics then result:

$$\begin{aligned} x_{i+1} &\approx x_i + \bar{\mu}(t_i, x_i) \Delta t + \bar{\sigma}(t_i, x_i) W(\Delta t) \\ &\quad + \frac{1}{2} \bar{\sigma}(t_i, x_i) (W^2(\Delta t) - \Delta t) \frac{\partial \bar{\sigma}}{\partial X}(t_i, x_i), \end{aligned}$$

where we used that $W(t_{i+1}) - W(t_i) \stackrel{d}{=} \sqrt{\Delta t} Z$, with $Z \sim \mathcal{N}(0, 1)$. This is the well-known Milstein discretization!

EXAMPLE: Milstein scheme for GBM

In the case of the ~~standard~~ GBM process, with $\bar{\mu}(t, S(t)) = \mu S(t)$ and $\bar{\sigma}(t, S(t)) = \sigma S(t)$, the discretization reads,

$$\begin{aligned}s_{i+1} &\approx s_i + \cancel{\mu s_i \Delta t} + \sigma s_i (W(t_{i+1}) - W(t_i)) \\&\quad + \frac{1}{2} \sigma^2 s_i ((W(t_{i+1}) - W(t_i))^2 - \Delta t) \\&\stackrel{d}{=} s_i + \cancel{\mu s_i \Delta t} + \sigma s_i \sqrt{\Delta t} Z + \frac{1}{2} \sigma^2 s_i (\Delta t Z^2 - \Delta t).\end{aligned}$$

This additional term improves the speed of convergence compared to the Euler scheme: for the GBM we get

- strong convergence of order 1 ;
- weak convergence of order 1 .

CODE \oplus **CODE** comparison.

SIMULATION OF THE CIR PROCESS

Standard discretization schemes, such as the Euler and Milstein schemes, are well-suited for simulating a wide range of stochastic processes. However, they typically do not perform well for processes that are, by definition, positive and have a probability mass around zero ~~and will blow up in the long run~~.

9.3.1 Challenges with standard discretization schemes

A typical example of a process with probability mass around zero is the CIR process, which was discussed to model the variance for the Heston stochastic volatility model in Chapter 8, with the following dynamics,

$$dv(t) = \kappa(\bar{v} - v(t))dt + \gamma\sqrt{v(t)}dW(t), \quad v(t_0) > 0. \quad (9.25)$$

Here :

- $\kappa \geq 0$ is the speed of mean reversion;
- $\bar{v} \geq 0$ is the long-term mean of the variance process;
- $\gamma > 0$ is the volatility of the volatility.

$$\text{Feller Condition: } 2k\bar{\sigma} > \gamma^2$$

- if it is satisfied, then the process $v(t)$ cannot reach zero;
- if it is not satisfied, then the origin is accessible but reflecting.

Hence in both cases, the process will cannot become negative.

The nonnegativity problem becomes apparent when a standard discretization is employed. If we apply, for example, the Euler discretization to the process in (9.25), i.e.,

$$v_{i+1} = v_i + \kappa(\bar{v} - v_i)\Delta t + \gamma\sqrt{v_i}\sqrt{\Delta t}Z_j$$

and assume that $v_i > 0$, we may calculate the probability that a next realization, v_{i+1} , becomes negative, i.e. $\mathbb{P}[v_{i+1} < 0]$:

$$\begin{aligned}\mathbb{P}[v_{i+1} < 0 | v_i > 0] &= \mathbb{P}[v_i + \kappa(\bar{v} - v_i)\Delta t + \gamma\sqrt{v_i\Delta t}Z < 0 | v_i > 0] \\ &= \mathbb{P}[\gamma\sqrt{v_i\Delta t}Z < -v_i - \kappa(\bar{v} - v_i)\Delta t | v_i > 0],\end{aligned}$$

which equals,

$$\mathbb{P}[v_{i+1} < 0 | v_i > 0] = \mathbb{P}\left[Z < -\frac{v_i + \kappa(\bar{v} - v_i)\Delta t}{\gamma\sqrt{v_i\Delta t}} | v_i > 0\right] > 0.$$

Since Z is a normally distributed random variable, it is unbounded. Therefore the probability of v_i becoming negative, is positive under the Euler discretization, implying $\mathbb{P}[v_{i+1} < 0 | v_i > 0] > 0$.

Why? The AR process is continuous, hence it can only become negative after first reaching zero. Notice that for $v|t|=0$, equation (9.25) becomes deterministic. As soon as the variance reaches zero, it will immediately be positive afterwards.

In contrast, the Euler discretization is not continuous.

Especially when the Feller condition is not satisfied most of the probability mass of the variance is concentrated around zero. This case gives rise to a high probability of the variance becoming negative when using the Euler discretization scheme. This is clearly undesirable. A similar exercise can be defined for the Milstein scheme.

Application of the Euler scheme to the CIR-type process may lead to undesired and unrealistic path realizations. Whereas the mean-reverting CIR process is guaranteed to be nonnegative, the Euler discretization is not.

Possible solutions :

- Truncated Euler scheme ;
- Reflecting Euler scheme ;
- Exact simulation .

Truncated Euler scheme

It can be summarized, as follows,

$$\begin{cases} \hat{v}_{i+1} = v_i + \kappa(\bar{v} - v_i)\Delta t + \gamma\sqrt{v_i\Delta t}Z, \\ v_{i+1} = \max(\hat{v}_{i+1}, 0). \end{cases}$$

In words: the paths that attain negative values are projected to the origin.

Notice that, by this truncation, a different process than the original CIR is represented numerically.

The accuracy of this scheme is parameter-dependent: if the Feller condition is not satisfied, then the density accumulate around zero and the adjusted paths may be highly biased.

Reflecting Euler scheme

The reflecting scheme is given by the following adjustment of the Euler scheme,

$$\begin{cases} \hat{v}_{i+1} = v_i + \kappa(\bar{v} - v_i)\Delta t + \gamma\sqrt{v_i\Delta t}Z, \\ v_{i+1} = |\hat{v}_{i+1}|. \end{cases} \quad (9.29)$$

In this case, the paths are forced to move upward. This is particularly useful when the Feller condition is not satisfied.

CODE

Exact simulation of the CIR model

We can use the fact that the variance process $v(t)$ follows a noncentral chi-squared distribution:

~~Distribution of $v(t)$~~) Process $v(t)|v(s)$, $t > s > 0$, under the CIR dynamics, is known to be distributed as $\bar{c}(t,s)$ times a noncentral chi-squared random variable, $\chi^2(\delta, \bar{\kappa}(t,s))$, where δ is the “degrees of freedom” parameter and the noncentrality parameter is $\bar{\kappa}(t,s)$, i.e.,

$$v(t)|v(s) \sim \bar{c}(t,s)\chi^2(\delta, \bar{\kappa}(t,s)), \quad t > s > 0, \quad (8.4)$$

with

$$\bar{c}(t,s) = \frac{1}{4\kappa}\gamma^2(1 - e^{-\kappa(t-s)}), \quad \delta = \frac{4\kappa\bar{v}}{\gamma^2}, \quad \bar{\kappa}(t,s) = \frac{4\kappa v(s)e^{-\kappa(t-s)}}{\gamma^2(1 - e^{-\kappa(t-s)})}. \quad (8.5)$$

Equation (8.44) may form the basis for an *exact simulation scheme* for the path realizations of the CIR process, as, for $i = 0, \dots, m - 1$,

$$\bar{c}(t_{i+1}, t_i) = \frac{\gamma^2}{4\kappa} \left(1 - e^{-\kappa(t_{i+1} - t_i)} \right),$$

$$\bar{\kappa}(t_{i+1}, t_i) = \frac{4\kappa e^{-\kappa(t_{i+1} - t_i)}}{\gamma^2(1 - e^{-\kappa(t_{i+1} - t_i)})} \boxed{v_i}.$$

$$\boxed{v_{i+1}} = \bar{c}(t_{i+1}, t_i) \chi^2(\delta, \bar{\kappa}(t_{i+1}, t_i)),$$

with a constant parameter $\delta = 4\kappa\bar{v}/\gamma^2$, and some initial value $v(t_0) = v_0$.

With this scheme, we can simulate CIR paths without paying special attention to the Feller condition.

The scheme relies however on an efficient sampling from the chi-squared distribution, and this may not be implemented efficiently in all packages.

System of SDEs

What if we need to simulate a system of SDEs?

This is the case, for example, of stochastic volatility models: here the volatility of the stock process is itself modelled by a diffusion process.

With an additional stochastic process, which is correlated to the asset price process $S(t)$, we deal with a system of SDEs.

Example: the Heston model

In the Heston SV model [Heston, 1993], we deal with two stochastic differential equations, one for the underlying asset price $S(t)$, and one for the variance process $v(t)$.

$$\begin{cases} dS(t) = \mu S(t)dt + \sqrt{v(t)}S(t)dW_x(t), & S(t_0) = S_0 > 0 \\ dv(t) = \kappa(\bar{v} - v(t))dt + \gamma\sqrt{v(t)}dW_v(t), & v(t_0) = v_0 > 0. \end{cases} \quad (8.18)$$

Here a correlation is defined between the underlying BMs, i.e

$$dW_n(t) dW_v(t) = \rho dt.$$

We recognize the AR process for the volatility!

of course, with the help of the Cholesky decomposition, the model can be presented in terms of independent BMs as:

$$\begin{cases} dS(t) = \mu S(t)dt + \sqrt{v(t)}S(t) \cancel{\left(\rho d\tilde{W}_v(t) + \sqrt{1-\rho^2} d\tilde{W}_x(t) \right)}, \\ dv(t) = \kappa(\bar{v} - v(t))dt + \gamma \sqrt{v(t)} \cancel{\left(\rho d\tilde{W}_x(t) + \sqrt{1-\rho^2} d\tilde{W}_v(t) \right)}, \end{cases}$$

with \tilde{W}_x and \tilde{W}_v independent.

Almost exact simulation of the Heston model

We now focus on the Heston stochastic volatility model, which, under the log transformation reads,

$$\begin{cases} dX(t) = \left(\mu - \frac{1}{2}v(t)\right) dt + \sqrt{v(t)} \left[\rho_{\text{v}} d\widetilde{W}_v(t) + \sqrt{1 - \rho_{\text{v}}^2} d\widetilde{W}_x(t) \right], \\ dv(t) = \kappa (\bar{v} - v(t)) dt + \gamma \sqrt{v(t)} d\widetilde{W}_v(t), \end{cases} \quad (9.59)$$

with the parameters as given for Equation (8.18).

After integration of both processes in (9.59) in a certain time interval $[t_i, t_{i+1}]$, the following discretization scheme is obtained:

$$x_{i+1} = x_i + \int_{t_i}^{t_{i+1}} \left(\mu - \frac{1}{2}v(t) \right) dt + \rho_{\text{v}} \boxed{\int_{t_i}^{t_{i+1}} \sqrt{v(t)} d\widetilde{W}_v(t)} \\ + \sqrt{1 - \rho_{\text{v}}^2} \int_{t_i}^{t_{i+1}} \sqrt{v(t)} d\widetilde{W}_x(t),$$

and

$$v_{i+1} = v_i + \kappa \int_{t_i}^{t_{i+1}} (\bar{v} - v(t)) dt + \gamma \boxed{\int_{t_i}^{t_{i+1}} \sqrt{v(t)} d\widetilde{W}_v(t)}.$$

Notice that the two integrals with $\widetilde{W}_v(t)$ in the SDEs above are the same, and in terms of the variance realizations they are given by:

$$\int_{t_i}^{t_{i+1}} \sqrt{v(t)} d\widetilde{W}_v(t) = \frac{1}{\gamma} \left(v_{i+1} - v_i - \kappa \int_{t_i}^{t_{i+1}} (\bar{v} - v(t)) dt \right). \quad (9.60)$$

As a final step in the Heston model simulation, the discretization for x_{i+1} is given by:

$$x_{i+1} = x_i + \int_{t_i}^{t_{i+1}} \left(\mu - \frac{1}{2} v(t) \right) dt + \frac{\rho_{\text{num}}}{\gamma} \left(v_{i+1} - v_i - \kappa \int_{t_i}^{t_{i+1}} (\bar{v} - v(t)) dt \right) \\ + \sqrt{1 - \rho_{\text{num}}^2} \int_{t_i}^{t_{i+1}} \sqrt{v(t)} d\widetilde{W}_x(t). \quad (9.61)$$

We approximate all integrals appearing in the expression above by their left integration boundary values of the integrand, as in the Euler discretization scheme:

$$x_{i+1} \approx x_i + \int_{t_i}^{t_{i+1}} \left(\mu - \frac{1}{2} v_i \right) dt + \frac{\rho_{\text{num}}}{\gamma} \left(v_{i+1} - v_i - \kappa \int_{t_i}^{t_{i+1}} (\bar{v} - v_i) dt \right) \\ + \sqrt{1 - \rho_{\text{num}}^2} \int_{t_i}^{t_{i+1}} \sqrt{v_i} d\widetilde{W}_x(t). \quad (9.62)$$

The calculation of the integrals is now trivial and results in:

$$x_{i+1} \approx x_i + \left(\mu - \frac{1}{2} v_i \right) \Delta t + \frac{\rho_{\text{num}}}{\gamma} (v_{i+1} - v_i - \kappa (\bar{v} - v_i) \Delta t) \\ + \sqrt{1 - \rho_{\text{num}}^2} \sqrt{v_i} (\widetilde{W}_x(t_{i+1}) - \widetilde{W}_x(t_i)). \quad (9.63)$$

After collecting all the terms and using that

$$\tilde{W}_n(t_{i+1}) - \tilde{W}_n(t_i) \stackrel{d}{=} \sqrt{\Delta t} Z_n \quad \text{with} \quad Z_n \sim N(0, 1),$$

we find :

$$x_{i+1} \approx x_i + \left(\mu - \frac{\rho}{\gamma} k \bar{v} \right) \Delta t + \left[\left(\frac{\rho k}{\gamma} - \frac{1}{2} \right) \Delta t - \frac{\rho}{\gamma} \right] v_i \\ + \frac{\rho}{\gamma} v_{i+1} + \sqrt{(1-\rho^2) \Delta t} \cdot \bar{v}_i$$

The variance v_{i+1} can be simulated, for a given value of v_i , by means of Euler scheme or by the noncentral chi-squared distribution. In this latter case, the algorithm is called "Almost exact simulation scheme".