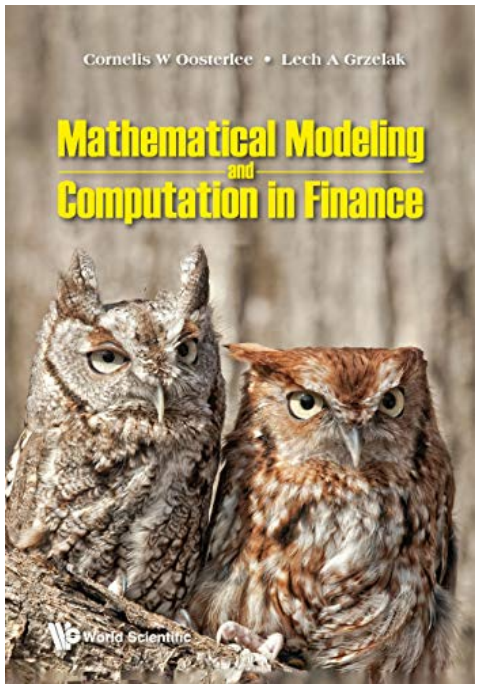
Materials for the course

The course is based on book "*Mathematical Modeling and Computation in Finance: With Exercises and Python and MATLAB Computer Codes*", by C.W. Oosterlee and L.A. Grzelak, World Scientific Publishing Europe Ltd, 2019. For more details go here.



- YouTube Channel with courses can be found here.
- Slides and the codes can be found here.

List of content

Monte Carlo Simulation[1]

- Monte Carlo methods were originally practiced under more generic names such as "statistical sampling".
- The name "Monte Carlo" was popularized by physics researchers S. Ulam, E. Fermi, J. von Neumann, and N. Metropolis.
- The name is a reference to a famous casino in Monaco where Ulam's uncle would borrow money to gamble.
- Perhaps the most famous early use was by Enrico Fermi in 1930, when he used a random method to calculate the properties of the newly-discovered neutron.
- Monte Carlo methods were central to the simulations required for the Manhattan Project, though were severely limited by the computational tools at the time.

[1] http://en.wikipedia.org/wiki/Monte_Carlo_method

Integration

- Monte Carlo Integration methods are sampling methods, based on probability theory.
- They rely on trials to reveal information.
- From an intuitive point of view, they rest on the Central Limit Theorem and the Law of Large Numbers.
- Monte Carlo methods are capable of handling quite complicated and large problems.
- Since the result of an experiment is a random number the structure of the error made has a probabilistic distribution.

Integration of a function

Suppose an $x - y$ plane, where we draw random numbers (dots in the graph). The MC integral of a function is approximately given by the total area times the fraction of points that fall under the curve $g(x)$.

- The greater the number of points the more accurate the evaluation of this area.
- Only competitive for complicated and/or multi-dimensional functions.

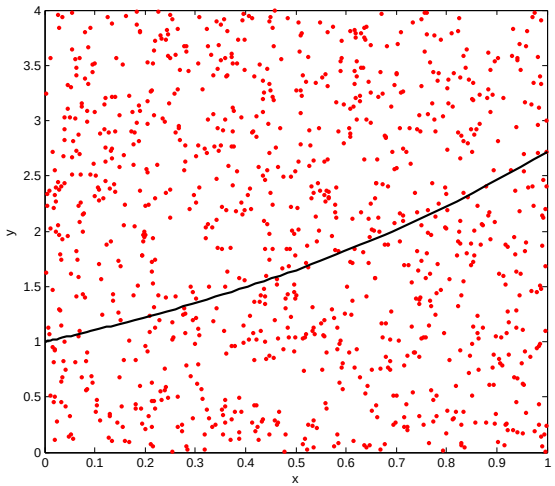


Figure: Basic idea of Monte Carlo integration, for $g(x) = \mathrm{e}^x$ over $[0, 1]$

- Suppose we evaluate

$$I = \int \int \int_{\Omega} g(x, y, z) \mathrm{d}x \mathrm{d}y \mathrm{d}z$$

- Choose a random point inside Ω. Value $\hat{g}$ of g at that point is an unbiased estimator of g inside Ω. An unbiased estimate of I is therefore $I = \hat{g} \times \Omega$.
- Although the estimator is unbiased, it has an outrageously large expected error (so it is essentially useless).
- Reduce the error by repeating the experiment lots of times, and average the results.
- Choose N random points, $I \approx (\hat{g}_1 + \hat{g}_2 + \ldots \hat{g}_N)\Omega_h/N$
- The expected error is reduced by a factor $\sqrt{N}$, so N should be chosen very large.
- The evaluation of integral will be better if the points are uniformly scattered in the entire area

Integration of a function

Crude application of Monte Carlo integration:

$$\int_0^1 \mathrm{e}^x \mathrm{d}x \approx 1.718281828459046$$

A huge number of data is needed to achieve, on average a good enough approximation ($10,000,000$ points are needed to reach an error of 6×10^{-5})

No.Samp.	$N = 10^1$	$N = 10^3$	$N = 10^4$	$N = 10^5$	$N = 10^7$
Est.	0.8000	1.5720	1.7448	1.7262	1.7182
error	0.9182	0.1462	-0.0265	-0.0079	0.0000614
time [s]	0.068	0.0689	0.0734	0.093	3.7609

Conclusion: It is clear that Monte Carlo integration without modifications is seldomly used for univariate integration.

Integration- Alternative approach

The alternative way of thinking about Monte Carlo integration is as follows:

$$\int_a^b g(x)\mathrm{d}x = (b-a)\mathbb{E}_{\mathcal{U}[a,b]}\left[g(x)\right],$$

which we can evaluate by drawing N random numbers, x_i, $i = 1, 2, \ldots, N$ from uniform distribution $\mathcal{U}[a, b]$, and calculating the following approximation:

$$\int_a^b g(x)\mathrm{d}x \approx \frac{b-a}{N}\sum_{i=1}^{N} g(x_i).$$

- Accuracy attained depends on the number of trials
- A key point is to get "good" random numbers.

Integration & Law of Large Numbers

The underlying idea of Monte Carlo integration is found in the Law of Large Numbers:

- If X_i is a collection of i.i.d. random variables with density $f(x)$, then:

$$\lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^{N} X_i = \int x f(x)\mathrm{d}x, \text{ a.s.}$$

- Further, we know that in this case:

$$\mathbb{V}\mathrm{ar}\left[\frac{1}{N}\sum_{i=1}^{N} X_i\right] = \frac{\sigma^2}{N}, \text{ where } \sigma^2 = Var(X_i),$$

When σ^2 is unknown, it can be estimated by an unbiased estimator:

$$\hat{\sigma}^2 = \frac{1}{N-1}\sum_{i=1}^{N}\left(X_i - \frac{1}{N}\sum_{i=1}^{N} X_i\right)^2.$$

Integration of a Function

- Naive approach

No.Samp.	$N = 10^1$	$N = 10^3$	$N = 10^4$	$N = 10^5$	$N = 10^7$
Est.	0.8000	1.5720	1.7448	1.7262	1.7182
error	0.9182	0.1462	-0.0265	-0.0079	0.0000614
time [s]	0.068	0.0689	0.0734	0.093	3.7609

- Expectation approach

$$\int_0^1 \mathrm{e}^x \mathrm{d}x = \mathbb{E}_{\mathcal{U}[0,1)}\left[\mathrm{e}^x\right],$$

No.Samp.	$N = 10^1$	$N = 10^3$	$N = 10^4$	$N = 10^5$	$N = 10^7$
Est.	1.7928	1.7126	1.7189	1.7188	1.7182
error	-0.0746	0.0056	-0.00071	-0.00053	0.0000119
time [s]	0.0776	0.0776	0.0763	0.0902	1.2859

- Let us check the Python code

Integrated Paths of Brownian Motion

- In finance we may encounter different types of stochastic integrals.

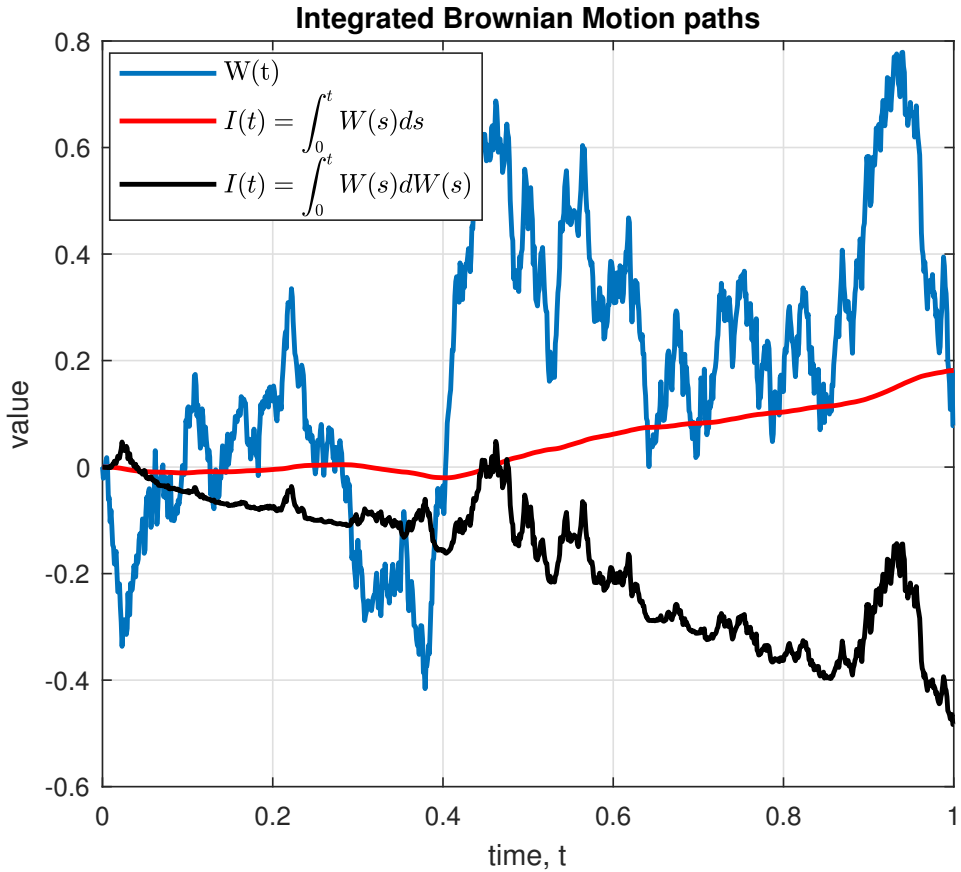


Figure: Stochastic paths for $W(t)$, $\int_0^t W(s)\mathrm{d}s$ and $\int_0^t W(s)\mathrm{d}W(s)$, as a function of running time t.

Exercise 1

- For a given deterministic function $g(t)$ determine:

$$G(T) = \int_0^T g(s)\mathrm{d}W(s)$$

First, define a partition: $0 = t_0 < t_1 < \cdots < t_M = T$. Then,

$$G(T) = \int_0^T g(s)\mathrm{d}W(s) \approx \sum_{k=0}^{M-1} g(t_k)\left(W(t_{k+1}) - W(t_k)\right).$$

- We take $g(t) = t^2$ and $T = 1$. Theoretically, we have:

$$\begin{aligned}
\mathbb{E}\left[\int_0^1 t^2\mathrm{d}W(t)\right] &= 0, \\
\mathbb{V}\mathrm{ar}\left[\int_0^1 t^2\mathrm{d}W(t)\right] &= \mathbb{E}\left[\int_0^1 t^2\mathrm{d}W(t)\right]^2 = \int_0^1 t^4\mathrm{d}t = 0.2
\end{aligned}$$

- Let's go to Python!

Exercise 2

- For a given Brownian motion $W(t)$ determine:

$$G(T) = \int_0^T W(s)\mathrm{d}W(s)$$

Analytically:

$$\mathbb{E}\left[\int_0^T W(s)\mathrm{d}W(s)\right] = 0$$

- In order to calculate the variance we define a function $F = x^2$, from Itô's lemma we have:

$$\mathrm{d}F = F_x\mathrm{d}x + \frac{1}{2}F_{xx}(\mathrm{d}x)^2 = 2x\mathrm{d}x + \frac{1}{2}\cdot 2(\mathrm{d}x)^2.$$

So,

$$\mathrm{d}W^2(t) = 2W(t)\mathrm{d}W(t) + (\mathrm{d}W(t))^2 = 2W(t)\mathrm{d}W(t) + \mathrm{d}t$$

Exercise 2, continuation

- Further, we have:

$$\begin{aligned} \int_0^T \mathrm{d}W^2(t) &= 2\int_0^T W(t)\mathrm{d}W(t) + \int_0^T \mathrm{d}t \\ W^2(T) - W^2(0) &= 2\int_0^T W(t)\mathrm{d}W(t) + T \end{aligned}$$

So,

$$\int_0^T W(t)\mathrm{d}W(t) = \frac{1}{2}W^2(T) - \frac{1}{2}T.$$

- For numerical computation we discretize:

$$G(T) = \int_0^T g(s)\mathrm{d}W(s) \approx \sum_{k=0}^{M-1} W(t_k)\left(W(t_{k+1}) - W(t_k)\right).$$

- **Example:** Let us take $T = 2$. Theoretically, we have:

$$\mathbb{E}\left[\int_0^2 W(s)\mathrm{d}W(s)\right] = 0,\ \mathbb{V}\mathrm{ar}\left[\int_0^2 W(s)\mathrm{d}W(s)\right] = 2.$$

- Let's go to Python!

Expectation and variance of $(W(t_{i+1}) - W(t))^2$

- Let us consider the following expectation, $\mathbb{E}\left[(\mathrm{d}W)^2\right]$. By the same steps as in the derivations of the Itô isometry we have:

$$\mathbb{E}\left[(\mathrm{d}W)^2\right] = \lim_{\Delta t \to 0} \mathbb{E}\left[(W(t+\Delta t) - W(t))^2\right] = \lim_{\Delta t \to 0} \Delta t = \mathrm{d}t, \quad (1)$$

and the variance is equal to:

$$\begin{aligned}
\mathbb{V}\mathrm{ar}\left[(\mathrm{d}W)^2\right] &= \lim_{\Delta t \to 0} \mathbb{V}\mathrm{ar}\left[(W(t+\Delta t) - W(t))^2\right] \\
&= \lim_{\Delta t \to 0} \mathbb{E}\left[(W(t+\Delta t) - W(t))^4\right] \\
&- \lim_{\Delta t \to 0} \left(\mathbb{E}\left[(W(t+\Delta t) - W(t))^2\right]\right)^2 \\
&= \lim_{\Delta t \to 0} 3(\Delta t)^2 - \lim_{\Delta t \to 0} (\Delta t)^2 = \lim_{\Delta t \to 0} 2(\Delta t)^2 = 2(\mathrm{d}t)^2.
\end{aligned}$$

- We conclude that the variance of $(\mathrm{d}W)^2$ converges to zero much faster than the expectation, when $\Delta t \to 0$. Because of this, we have as a stochastic calculus rule,

$$(\mathrm{d}W)^2 = \mathrm{d}t,$$

as the variance approaches zero rapidly in the limit.

Expectation and variance of $(W(t_{i+1}) - W(t))^2$

- We gave a heuristic argument for the fact that $\mathrm{d}W^2 = \mathrm{d}t$ can be used. Here, we perform a numerical experiment, in which we measure the expectation and variance of the term $(W(t_{i+1}) - W(t_i))^2$, with respect to a time discretization.

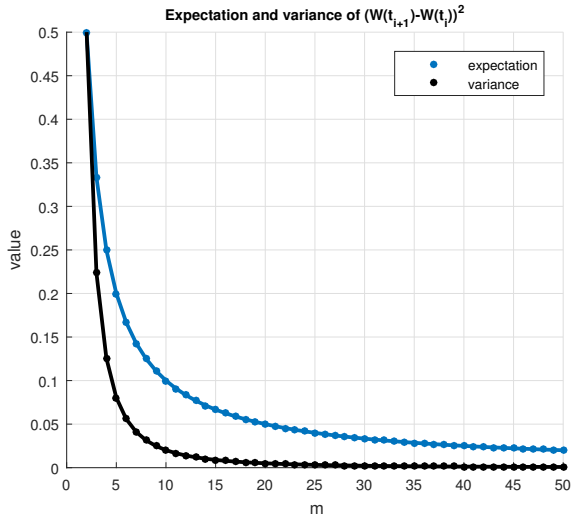


Figure: Expectation and variance in the numerical approximation of $\mathrm{d}W^2$ versus parameter m.

- The results in Figure confirm that the variance converges to 0 much more rapidly than the expectation, and that the expectation converges to $\mathrm{d}t$.

Smoothness of a Payoff and Impact on Convergence

- In this example we will show that the Monte Carlo convergence strongly depends on the smoothness of the function under consideration.
- Let's take the following two (nonsmooth and smooth) functions,

$$g_1(x) = \mathbb{1}_{x \geq 0}, \quad \text{and} \quad g_2(x) = F_{\mathcal{N}(0,1)}(x),$$

and consider the following expectations, $\mathbb{E}[g_i(W(1))|\mathcal{F}(t_0)]$, for $i = 1, 2$, where $W(1)$ is a Brownian motion at time $T = 1$. Both expectations can be solved analytically and are equal to $\frac{1}{2}$:

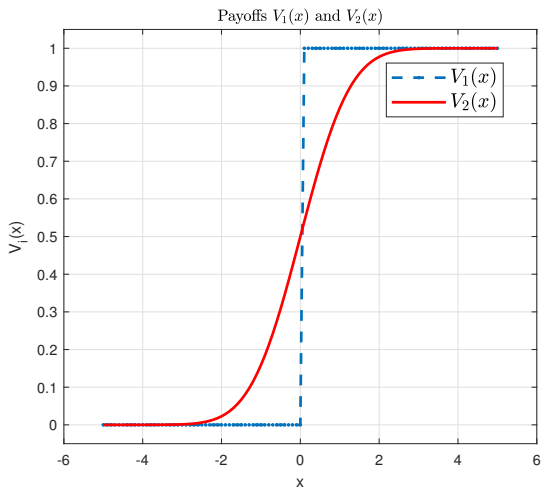
$$\begin{aligned} V_1 := \mathbb{E}[g_1(W(1))] &= \int_{\mathbb{R}} \mathbb{1}_{x \geq 0} f_{\mathcal{N}(0,1)}(x) = \frac{1}{2}, \\ V_2 := \mathbb{E}[g_2(W(1))] &= \int_{\mathbb{R}} F_{\mathcal{N}(0,1)}(x) f_{\mathcal{N}(0,1)}(x) \mathrm{d}x = \int_{\mathbb{R}} F_{\mathcal{N}(0,1)}(x) \mathrm{d}F(x) \\ &= \frac{1}{2} F^2_{\mathcal{N}(0,1)}(x) \Big|_{-\infty}^{+\infty} = \frac{1}{2}. \end{aligned}$$

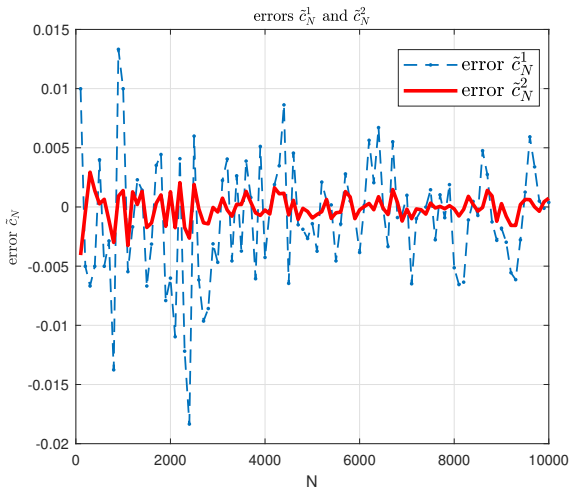
Smoothness of a Payoff and Impact on Convergence

- We define the following error,

$$\widetilde{c}_N^i = \widetilde{V}_i^N - V_i, \quad \text{with} \quad \widetilde{V}_i^N = \frac{1}{N}\sum_{j=1}^{N} V_i(w_j),$$

where w_j are samples from $W(1) \sim \mathcal{N}(0,1)$, and the exact solution $V_i = 1/2$.





- The results in Figure confirm that functions that are of *digital option type*, thus are nonsmooth, may require more paths to achieve the same level of accuracy than for smooth functions.

Strong and weak convergence

Definition (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, $t_i = i \cdot T/m$, $i = 0, \ldots, m$. Then, the approximation x_m converges in a strong sense *to $X(T)$, with order $\alpha > 0$, if*

$$\epsilon^s(\Delta t) := \mathbb{E}^{\mathbb{Q}}\left[|x_m - X(T)|\right] = O(h^{\alpha}).$$

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*

$$\epsilon^w(\Delta t) := \left|\mathbb{E}^{\mathbb{Q}}[g(x_m)] - \mathbb{E}^{\mathbb{Q}}[g(X(T))]\right| = O(h^{\beta}).$$

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 $X(T)$.*

MC option valuation

- Today's value of a European-style derivative is given, via the Feynman-Kac Theorem, by

$$\begin{aligned} V(t_0, S_0) &= \mathrm{e}^{-r(T-t_0)}\mathbb{E}^{\mathbb{Q}}\left[H(T,S)\middle|\mathcal{F}(t_0)\right] \\ &= \mathrm{e}^{-r(T-t_0)}\int_{\mathbb{R}} H(T,y) f_S(T,y;t_0,S_0)\mathrm{d}y, \end{aligned}$$

  where $S_0 = S(t_0)$ and the expectation $\mathbb{E}^{\mathbb{Q}}[\cdot]$ is taken under risk-neutral measure $\mathbb{Q}$, and $f_S(T,y;t_0,S_0)$ represents the probability density function, connected to the general risk-neutral Itô dynamics,

$$\mathrm{d}S(t) = \bar{\mu}^{\mathbb{Q}}(t,S)\mathrm{d}t + \bar{\sigma}(t,S)\mathrm{d}W^{\mathbb{Q}}(t), \quad t > t_0. \tag{2}$$

- When the PDF is not available in closed-form, the Monte Carlo method is a convenient and valuable pricing method. By approximation and simulation of the asset dynamics, asset prices at time T, giving an estimation of the PDF by means of a histogram.

Monte Carlo algorithm

1. Partition the time interval $[0, T]$, $0 = t_0 < t_1 < \cdots < t_m = T$.
2. Generate asset values, $s_{i,j}$, taking the risk-neutral dynamics of the underlying model. $s_{i,j}$ has two indices, the time points and the Monte Carlo path.
3. Compute the N *payoff values*, H_j. In the case of European options, $H_j = H(T, s_{m,j})$, in the case of path-dependent options, $H_j = H(T, s_{i,j})$, $i = 1, \ldots m$.
4. Compute the average, $\mathbb{E}^{\mathbb{Q}}\left[H(T,S)\middle|\mathcal{F}(t_0)\right] \approx \frac{1}{N}\sum_{j=1}^{N} H_j =: \bar{H}_N$.
5. Calculate the option value as $V(t_0, S) \approx \mathrm{e}^{-r(T-t_0)}\frac{1}{N}\sum_{j=1}^{N} H_j$.
6. Determine the standard error related to the obtained prices in Step 5.

Standard error

- By the strong law of large numbers, we know that, for $N \to \infty$,

$$\lim_{N\to\infty} \bar{H}_N(T,S) = \mathbb{E}^{\mathbb{Q}}[H(T,S)], \quad \text{with probability} \quad 1.$$

- To estimate the error due to a finite number of paths, we compute,

$$\begin{aligned}\mathbb{V}\text{ar}^{\mathbb{Q}}\left[\bar{H}_N(T,S)\right] &= \mathbb{V}\text{ar}^{\mathbb{Q}}\left[\frac{1}{N}\sum_{j=1}^{N} H(T,s_{m,j})\right] \\ &= \frac{1}{N^2}\sum_{j=1}^{N}\mathbb{V}\text{ar}^{\mathbb{Q}}[H(T,s_{m,j})] \approx \frac{1}{N}\mathbb{V}\text{ar}^{\mathbb{Q}}[H(T,S)],\end{aligned}$$

  given that samples $s_{m,j}$ are drawn independently.
- The unknown variance is approximated by the sample variance,

$$\bar{v}_N^2 := \frac{1}{N-1}\sum_{j=1}^{N}\left(H(T,s_{m,j}) - \bar{H}_N(T,S)\right)^2.$$

- Standard error ϵ_N is defined as $\epsilon_N := \frac{\bar{v}_N}{\sqrt{N}}$. When the number of samples increases by a factor 4, the error reduces by a factor 2.

Euler Discretization

- $S(t)$ denotes a stochastic process and solution of an SDE,

$$\mathrm{d}S(t) = \bar{\mu}(S(t), t)\mathrm{d}t + \bar{\sigma}(S(t), t)\mathrm{d}W(t) \text{ for } 0 \leq t \leq T.$$

where the driving process $W(t)$ is a Wiener process. We already know the Euler discretization:

$$\begin{cases} s_{i+1} &= s_i + \bar{\mu}(s_i, t_i)\Delta t + \bar{\sigma}(s_i, t_i)\Delta W_i \ , \ t_i = i\Delta t \\ \Delta W_i &= W_{i+1} - W_i = Z\sqrt{\Delta t} \text{ with } Z \sim N(0,1) \end{cases} \tag{3}$$

where length $h \equiv \Delta t$ is assumed equidistant, i.e., $h \equiv \Delta t = \frac{T}{M}$.

Definition (Absolute error)

The absolute error at time T is defined as:

$$\epsilon(h) := \mathbb{E}\left(|S_T - s_T^h|\right).$$

Approximation Error

- Within the Monte Carlo simulation, with many paths, the Euler discretization for the i^{th} timestep and j^{th} path, reads:

$$s_{i+1,j} \approx s_{i,j} + r s_{i,j} \Delta t + \sigma s_{i,j} \left(W_{i+1,j} - W_{i,j} \right), \tag{4}$$

  with $\Delta t = t_{i+1} - t_i$, for any $i = 1, \ldots m$, $s_0 = S(t_0)$ and $j = 1, \ldots N$.
- The GBM process with dynamics, $\mathrm{d}S(t) = rS(t)\mathrm{d}t + \sigma S(t)\mathrm{d}W(t)$, has as exact solution in the time interval $[t_i, t_{i+1}]$,

$$S(t_{i+1}) = S(t_i) \exp\left((r - \frac{1}{2}\sigma^2)\Delta t + \sigma \left(W(t_{i+1}) - W(t_i) \right) \right). \tag{5}$$

- As an example, we set $S(t_0) = 50$, $r = 0.06$, $\sigma = 0.3$, $T = 1$ and determine the strong convergence error at the maturity time T, i.e.

$$\epsilon^s(\Delta t) = \frac{1}{N} \sum_{j=1}^{N} |S_j(T) - s_{m,j}| = \frac{1}{N} \sum_{j=1}^{N} |S(t_0) \mathrm{e}^{\left(r - \frac{1}{2}\sigma^2\right)T + \sigma W_{m,j}} - s_{m,j}|,$$

Approximation Error

- For different Δt-values, and also the weak convergence error:

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

  where m stands for time $t_m \equiv T$ and the index j indicates the path number at which solutions from (5) and (4) are evaluated.

- Note that in this experiment it is crucial to use the *same Brownian motion* for both equations (5) and (4). If the Brownian motions wouldn't be the same, we wouldn't be able to measure the strong convergence, as the random paths would be different.

Approximation Error

- We postulate that

$$\epsilon^s(\Delta t) \leq C \cdot (\Delta t)^{\frac{1}{2}} = \mathcal{O}((\Delta t)^{\frac{1}{2}}).$$

- For different mesh widths Δt, the results are presented in Figure below.

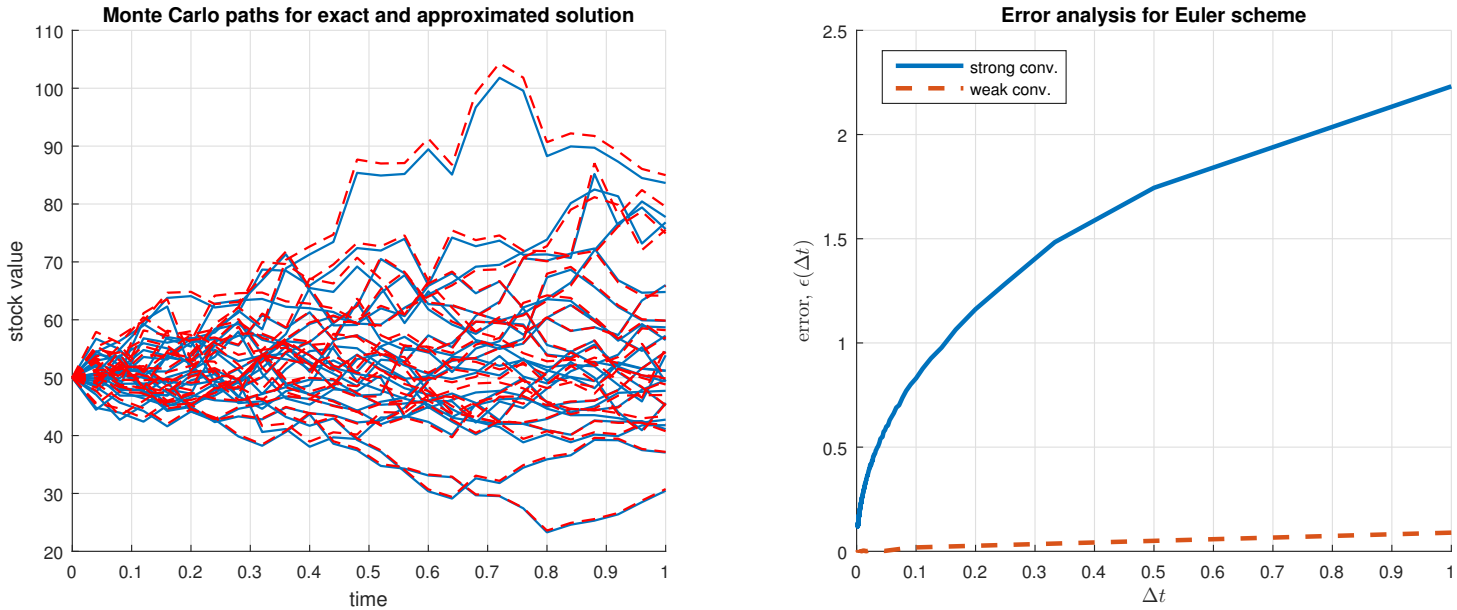


Figure: Left: generated paths with exact simulation (5) versus Euler approximation (4); Right: error against value of time step Δt for the Euler discretization.

Milstein Discretization

- In the case of deterministic differential equations, one may employ the *Taylor expansion* to define discretizations by which we may obtain a higher order of convergence. For stochastic differential equations a similar approach is available, which is based on the stochastic Taylor expansion, or the so-called *Itô-Taylor expansion*. The stochastic Euler approximation is based on the first two terms of this expansion.
- For the Itô process SDE, $\mathrm{d}X(t) = \bar{\mu}(t, X(t))\mathrm{d}t + \bar{\sigma}(t, X(t))\mathrm{d}W(t)$, the discretization under the *Milstein scheme* is obtained by adding a *third term* to the Euler discretization, i.e.,

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

with $x_0 = X(t_0)$.

Milstein Discretization

- In the case of the risk neutral GBM process, with $\bar{\mu}^{\mathbb{Q}}(t, S(t)) = rS(t)$ and $\bar{\sigma}(t, S(t)) = \sigma S(t)$, the discretization reads,

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

- The additional correction term in the Milstein scheme improves the *speed of convergence* compared to the Euler discretization for scalar SDEs. For the Black-Scholes model, as well as for the local volatility model, this scheme exhibits both a strong and weak convergence of order 1.
- Although the Milstein Scheme is definitely manageable in the one-dimensional case, *its extension to multi-dimensional SDE problems is far from trivial*.

Milstein Discretization

- We set again, $S(t_0) = 50$, $r = 0.06$, $\sigma = 0.3$, $T = 1$, and measure the strong and weak convergence.

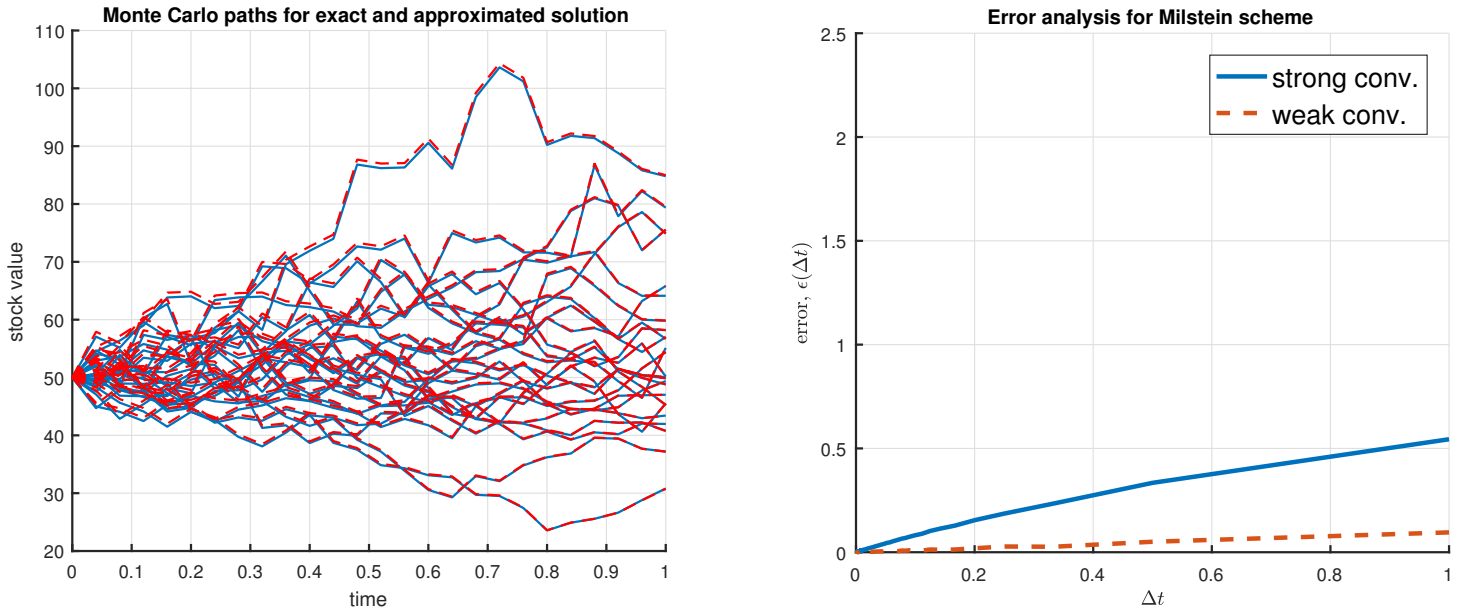


Figure: Left: generated paths exact vs. approximation; Right: error against value of step size Δt for the Milstein discretization

- Based on these results, we confirm that

$$\boxed{\epsilon^s(\Delta t) \leq C \cdot \Delta t = \mathcal{O}(\Delta t),}$$

for the Milstein scheme in this test.

Milstein vs. Euler Discretization

- Error comparison between Euler and Milstein.

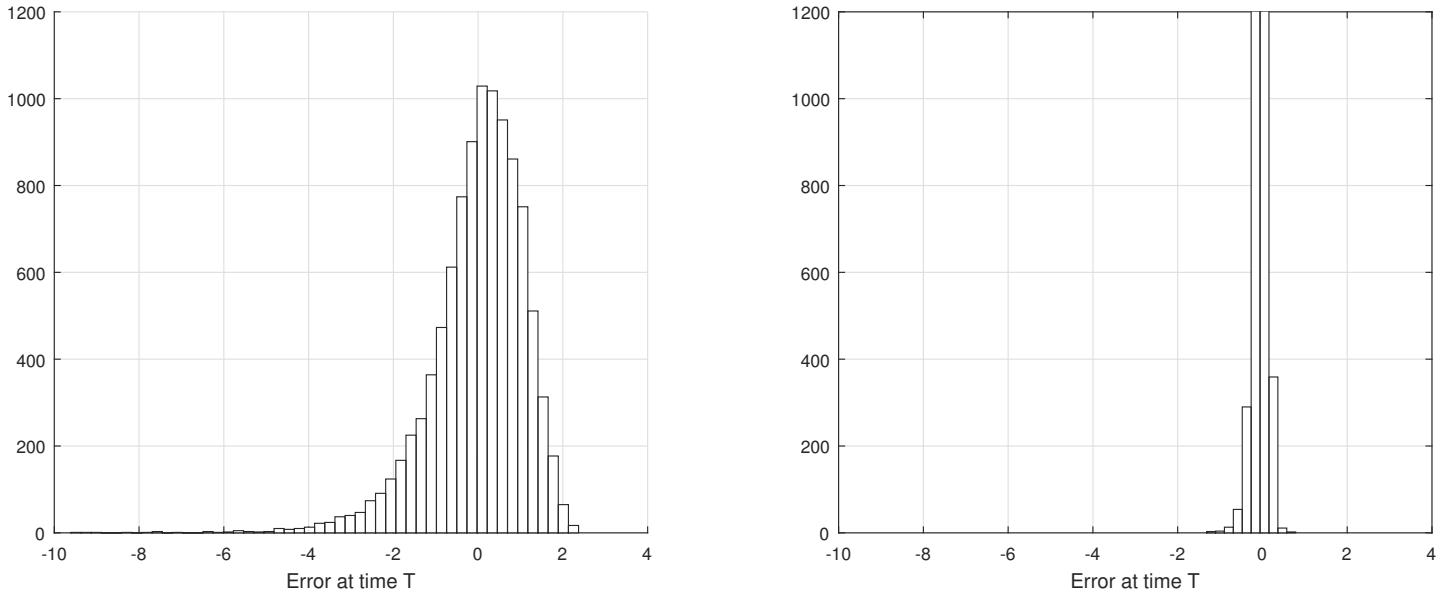


Figure: Error comparison between Euler (left) and Milstein (right) discretization schemes, for $T = 1$ with $\Delta t = 0.1$.