Simulation of diffusion processes *

Hiroki Masuda

Kyushu University
JST CREST

YUIMA Summer School 2019
Brixen-Bressanone, Italy
June 25–28, 2019

* This version: June 26, 2019

- **Diffusion**, widely used in: finance, biology, life sciences, ecology, epidemiology, microelectronics, engineering, mechanics, chemistry, ...

- This class: a concise overview of approximate generation of diffusions.

Contents

1. Why approximation (discretization)?
2. Euler-Maruyama scheme

Kloeden, P. E. and Platen, E. (1992).
Numerical solution of stochastic differential equations, volume 23 of *Applications of Mathematics (New York)*.
Springer-Verlag, Berlin.

Maruyama, G. (1955).
Continuous Markov processes and stochastic equations.
Rend. Circ. Mat. Palermo (2), 4:48–90.

1. Why approximation (discretization)?

2. Euler-Maruyama scheme

Diffusion process

- A solution to the d-dimensional stochastic differential equation (SDE):

$$dX_t = a(t, X_t)dt + b(t, X_t)dw_t, \qquad t \in \mathbb{R}_+ := [0, \infty), \tag{1}$$

or in the integral form

$$X_t = X_s + \int_s^t a(u, X_u)du + \int_s^t b(u, X_u)dw_u, \quad t > s.$$

User's input for simulating $X = (X_t)$ on computer:

- Initial variable $X_0 \in \mathbb{R}^d$, possibly random.
- r-dimensional standard Wiener process $w = (w^j)_{j=1}^r$, independent of X_0
- **Drift** coefficient $a(t, x) = \{a_k(t, x)\}_{k \le d} : \mathbb{R}_+ \times \mathbb{R}^d \to \mathbb{R}^d$
- **Diffusion** coefficient $b(t, x) = \{b_{kl}(t, x)\}_{k \le d;\, l \le r} : \mathbb{R}_+ \times \mathbb{R}^d \to \mathbb{R}^d \otimes \mathbb{R}^r$

Coefficients

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

- The coefficients are smooth in t and **globally Lipschitz** in x:

$$\exists C > 0,\ \forall t \in \mathbb{R}_+,\ \forall x_1, x_2 \in \mathbb{R}^d,$$
$$|a(t, x_1) - a(t, x_2)| + |b(t, x_1) - b(t, x_2)| \le C|x_1 - x_2|. \qquad (2)$$

- (2) entails the **linear-growth** property if $\sup_t |a \vee b|(t, 0) < \infty$:

$$\exists C' > 0,\ \forall t \in \mathbb{R}_+,\ \forall x \in \mathbb{R}^d,$$
$$|a(t, x)| + |b(t, x)| \le C'(1 + |x|).$$

Then (1) admits a unique non-explosive continuous Markov solution.

- Even for the ODE

$$dX_t = a(t, X_t)dt, \quad X_0 = x_0,$$

one needs care:

  - Non-linear-growth, *explosive* (in finite time) example:

$$dx_t = x_t^2 dt, \quad x_0 > 0,$$

    the solution being $x_t = x_0/(1 - x_0 t)$.

  - Non-Lipschitz, *non-unique* example:

$$dx_t = |x_t|^\alpha \quad (0 < \alpha < 1), \quad x_0 = 0,$$

    and for any $t_0 > 0$, the process given by $x_t = 0$ ($t \in [0, t_0]$) and $x_t = \{(1-\alpha)(t - t_0)\}^{1/(1-\alpha)}$ ($t > t_0$) is a solution.

Discrete-time sample

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

$$X_t = X_s + \int_s^t a(u, X_u)du + \int_s^t b(u, X_u)dw_u, \quad t > s.$$

- Our primary interest here is to generate a *discrete-time* sample

$$\boldsymbol{X}_n := (X_{t_0}, X_{t_1}, \dots, X_{t_n}), \tag{3}$$

  where $t_j = jh$ with a sampling stepsize $h > 0$.

- If we can generate X_h given any $h > 0$ and $X_0 = x_0$:

$$X_h = x_0 + \int_0^h a(s, X_s)ds + \int_0^h b(s, X_s)dw_s,$$

  then, by the Markov nature of X, we may **inductively** generate (3).

Unfortunately, the SDE (1) cannot be explicitly solved except for some particular cases, thus necessitating some approximation.

Particular example: Gaussian additive process

- For a *non-random* function (a, b) of time, smooth enough, the process

$$X_t = X_0 + \int_0^t a_s ds + \int_0^t b_s dw_s$$

defines a Gaussian Markov process **with independent increments**:

$$X_t - X_s = \int_s^t a_u du + \int_s^t b_u dw_u \sim N\left(\int_s^t a_u du,\ \int_s^t b_u^{\otimes 2} du\right),$$

where $b_u^{\otimes 2} := b_u b_u^\top \in \mathbb{R}^d \otimes \mathbb{R}^d$.

- The Gaussianity of the *Wiener integral*: roughly,

$$\int_0^h b_u dw_u \approx \sum_{k=1}^m b_{(k-1)h/m} \left(w_{kh/m} - w_{(k-1)h/m}\right)$$

$$\overset{\mathcal{L}}{\approx} N\left(0,\ \frac{h}{m}\sum_{k=1}^m b_{(k-1)h/m}^{\otimes 2}\right) \overset{\mathcal{L}}{\approx} N\left(0,\ \int_0^h b_u^{\otimes 2} du\right), \quad m \to \infty.$$

Particular example: Gaussian Ornstein-Uhlenbeck process

- Univariate continuous-time AR(1) ($\lambda \in \mathbb{R}$, $\sigma > 0$):

$$dX_t = -\lambda X_t dt + \sigma dw_t$$

admits an explicit solution (by Itô's formula)

$$X_t = e^{-\lambda t} X_0 + \sigma \int_0^t e^{-\lambda(t-s)} dw_s.$$

- Exact simulation is simple enough:

$$\begin{aligned}\mathcal{L}\left(X_h \middle| X_0 = x_0\right) &= N\left(e^{-\lambda t} x_0,\, \sigma^2 \int_0^h e^{-2\lambda(h-s)} ds\right) \\ &= N\left(e^{-\lambda t} x_0,\, \frac{\sigma^2}{2\lambda}(1 - e^{-2\lambda h})\right).\end{aligned}$$

- More generally, (λ, σ) could be time-varying and random:

$$X_t = \exp\left(-\int_s^t \lambda_u du\right) X_s + \int_s^t \exp\left(-\int_u^t \lambda_v dv\right) \sigma_u dw_u$$

1 Why approximation (discretization)?

2 Euler-Maruyama scheme

Discretization formula

Needs to approximate the time (Riemann) and stochastic integrals:

$$X_t = X_s + \int_s^t a(u, X_u)du + \int_s^t b(u, X_u)dw_u.$$

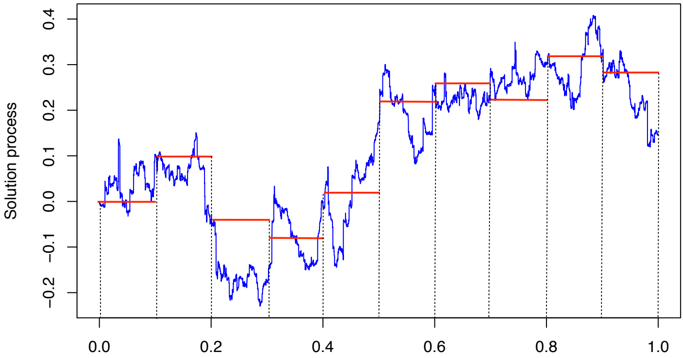
- **Euler(-Maruyama) scheme** with discretization stepsize $\Delta > 0$:

$$\int_t^{t+\Delta} a(s, X_s)ds \approx \int_t^{t+\Delta} a(t, X_t)ds = a(t, X_t)\Delta$$
$$\int_t^{t+\Delta} b(s, X_s)dw_s \approx \int_t^{t+\Delta} b(t, X_t)dw_s = b(t, X_t)(w_{t+\Delta} - w_t)$$

  - Typically, the approximation errors are $O_p(\Delta^{3/2})$ and $O_p(\Delta)$ for $\Delta \to 0$, respectively.

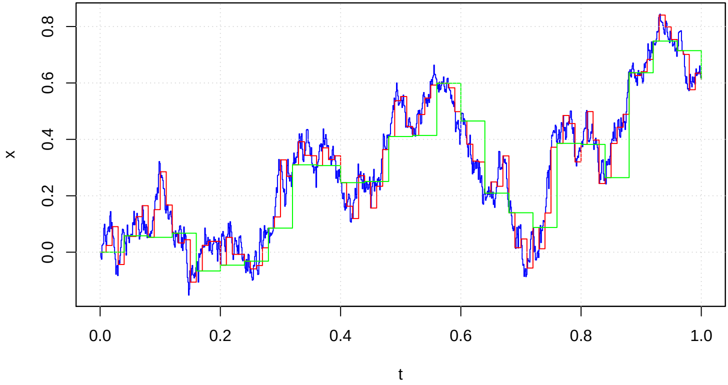
Euler-approximation process (càdlàg and piecewise constant):

$$X_t^\Delta := X_{\Delta[t/\Delta]}$$
$$= \begin{cases} X_0 & t \in [0,\Delta), \\ X^\Delta_{(j-1)\Delta} + a\left((j-1)\Delta,\, X^\Delta_{(j-1)\Delta}\right)\Delta & \\ \quad +b\left((j-1)\Delta,\, X^\Delta_{(j-1)\Delta}\right)\left(w_{j\Delta} - w_{(j-1)\Delta}\right) & t \in [(j-1)\Delta, j\Delta). \end{cases} \tag{4}$$



YUIMA internally runs the steps through `simulate` function:

1. generates a finest-approximating process X^{Δ},
2. subsamples a *thinned* data $X^{k\Delta}$ for an integer $k \geq 2$,
3. obtains an approximate $X_0, X_h, X_{2h}, \ldots, X_{nh}$, with stepsize $h := k\Delta$.

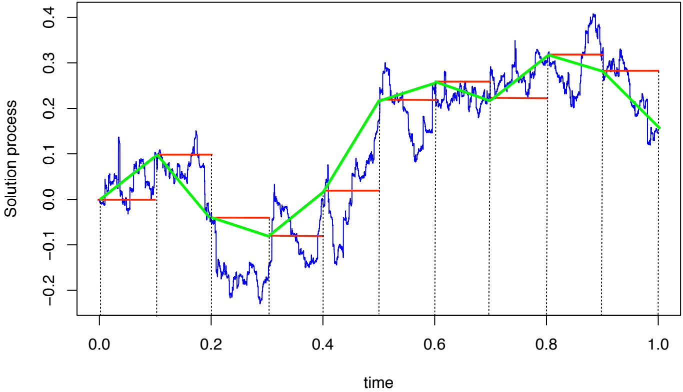


- Euler scheme (4) approximately embodies **strong solution** to (1):

$$X_{N\Delta} \approx X^{\Delta}_{N\Delta} =: F_{N,\Delta}\left(X_0,\, \{w_{j\Delta} - w_{(j-1)\Delta}\}_{j=1}^{N}\right)$$

  for some $F_{N,\Delta}$;
  - X^{Δ}_{Δ} is a functional of X_0 and w_Δ (with time variable t),
  - $X^{\Delta}_{2\Delta}$ is a functional of X_0 and $\{w_\Delta, w_{2\Delta} - w_\Delta\}$,
  - ...
  - $X^{\Delta}_{N\Delta}$ is a functional of X_0 and $\{w_\Delta, w_{2\Delta} - w_\Delta, \dots, w_{N\Delta} - w_{(N-1)\Delta}\}$.

- polygonal-line approximation [Maruyama, 1955].
  - May be useful when arguing weak convergence of processes.



- Another popular improvement is the **Milstein scheme**.

Approximation errors

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

- X^Δ is a *strong approximation of order* $\gamma > 0$ if

$$\mathbb{E}\left(|X_t^\Delta - X_t|\right) \le C_t \Delta^\gamma.$$

- Given an f smooth enough, X^Δ is a *weak approximation of order* $\beta > 0$ if

$$\left|\mathbb{E}\{f(X_t)\} - \mathbb{E}\{f(X_t^\Delta)\}\right| \le C_t \Delta^\beta.$$

- As for the Euler scheme, $\gamma = 0.5$ and $\beta = 1$ (not for free!).
- As a matter of fact, the *locally uniform* approximation does hold:

$$\forall T > 0, \quad \lim_{\Delta \to 0} \mathbb{E}\left(\sup_{t \in [0,T]} \left|X_t - X_t^\Delta\right|\right) = 0.$$

  - e.g. [Kloeden and Platen, 1992, theorem 10.2.2]

Concluding remarks

$$X_t = X_s + \int_s^t a(u, X_u)du + \int_s^t b(u, X_u)dw_u, \quad t > s.$$

$$X_t^\Delta := X_{\Delta[t/\Delta]}$$
$$= \begin{cases} X_0 & t \in [0, \Delta), \\ X^\Delta_{(j-1)\Delta} + a\left((j-1)\Delta,\, X^\Delta_{(j-1)\Delta}\right)\Delta \\ \quad + b\left((j-1)\Delta,\, X^\Delta_{(j-1)\Delta}\right)\left(w_{j\Delta} - w_{(j-1)\Delta}\right) & t \in [(j-1)\Delta, j\Delta). \end{cases}$$

- YUIMA can flexibly generate the Euler-approximation process X^Δ.
- Theoretical error bounds: the smaller Δ is, the better the approximations.
- Nevertheless, we need to take care about **regularity of the coefficients**!

■