CS-E5885 Modeling biological networks
Diffusion processes

Harri Lähdesmäki

Department of Computer Science
Aalto University

January 21, 2022

Outline

- Diffusion processes
- Stochastic differential equations
- Chemical Langevin equation

- Reading (see references at the end):
  - This lecture follows closely Sections 5.5 and 8.3 from (Wilkinson, 2011)

Diffusion processes

- Continuous-time Markov chain with continuous state space are called diffusion processes
- Motivation:
  - Diffusion processes provide a good approximation to biochemical reaction networks
  - Continuous state space models are easier to work with
- This lecture covers a non-technical introduction to diffusion processes

Brownian motion

- A univariate Brownian motion B is a continuous-time process defined for $t > 0$ as follows
  1. $B_0 = 0$
  2. $B_t - B_s \sim N(0, t - s)$, $\forall t > s$
  3. The increment $B_t - B_s$ is independent of the increment of a non-overlapping time interval $B_{t'} - B_{s'}$, $\forall t > s \geq t' > s'$

Brownian motion

- A univariate Brownian motion B is a continuous-time process defined for $t > 0$ as follows
  1. $B_0 = 0$
  2. $B_t - B_s \sim N(0, t - s), \forall t > s$
  3. The increment $B_t - B_s$ is independent of the increment of a non-overlapping time interval $B_{t'} - B_{s'}$, $\forall t > s \geq t' > s'$
- From property 2, we see that
  - $B_t \sim N(0, t)$
  - If for a small time increment Δt we define process increment $\Delta B = B_{t+\Delta t} - B_t$, then

$$\Delta B \sim N(0, \Delta t)$$

  - Process increment for a small time increment provides a simulation method for Brownian motion (at fixed time points)

$$B_0 = 0, \quad B_{\Delta t} = \Delta B^{(1)}, \quad B_{2\Delta t} = B_{\Delta t} + \Delta B^{(2)}, \quad B_{3\Delta t} = B_{2\Delta t} + \Delta B^{(3)}, \ldots$$

where $\Delta B^{(i)}$ values denote realizations of the Gaussian-distributed process increment ΔB at each time increment

Brownian motion illustration

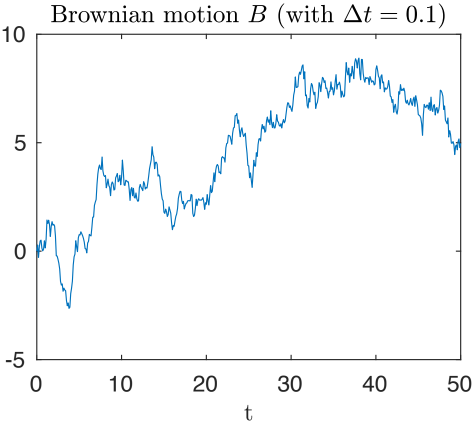

Figure: Brownian motion illustration

Diffusion process in 1-dimension

- A 1-dimensional Itô diffusion process X_t is governed by a stochastic differential equation (SDE) of the form
$$dX_t = \underbrace{\mu(X_t)dt}_{\text{deterministic}} + \underbrace{\sigma(X_t)dB_t}_{\text{stochastic}}$$

Diffusion process in 1-dimension

- A 1-dimensional Itô diffusion process X_t is governed by a stochastic differential equation (SDE) of the form

$$dX_t = \underbrace{\mu(X_t)dt}_{\text{deterministic}} + \underbrace{\sigma(X_t)dB_t}_{\text{stochastic}}$$

- SDE interpretation is that
  1. The process is Markovian
  2. The infinitesimal mean of the process is $\mu(X_t)$
  3. The infinitesimal variance of the process is $\sigma^2(X_t)$

Diffusion process approximation

- In 1-D, given the value of X_t, we can see easily that the process dynamics

$$dX_t = \underbrace{\mu(X_t)dt}_{\text{deterministic}} + \underbrace{\sigma(X_t)dB_t}_{\text{stochastic}}$$

for a small time increment Δt are distributed approximately as

$$\Delta X_t = X_{t+\Delta t} - X_t \sim N(\mathrm{E}(\Delta X_t), \mathrm{Var}(\Delta X_t)),$$

where

$$\mathrm{E}(\Delta X_t) = \mu(X_t)\Delta t \quad \text{and} \quad \mathrm{Var}(\Delta X_t) = \sigma^2(X_t)\Delta t$$

Diffusion approximation example

- Recall the immigration-death process from Lecture 2 (Wilkinson, 2011, Section 5.4.3)
- For an infinitesimal time increment dt (when the state at time t is $x \in \mathbb{Z}_{\geq 0}$)

$$\begin{aligned} P(X_{t+dt} = x - 1) &= x\mu dt \\ P(X_{t+dt} = x) &= 1 - (\lambda + x\mu)dt \\ P(X_{t+dt} = x + 1) &= \lambda dt \end{aligned}$$

Diffusion approximation example

- Recall the immigration-death process from Lecture 2 (Wilkinson, 2011, Section 5.4.3)
- For an infinitesimal time increment dt (when the state at time t is $x \in \mathbb{Z}_{\geq 0}$)

$$\begin{aligned} P(X_{t+dt} = x - 1) &= x\mu dt \\ P(X_{t+dt} = x) &= 1 - (\lambda + x\mu)dt \\ P(X_{t+dt} = x + 1) &= \lambda dt \end{aligned}$$

- The same equations hold for the process updates

$$\begin{aligned} P(dX_t = -1) &= x\mu dt \\ P(dX_t = 0) &= 1 - (\lambda + x\mu)dt \\ P(dX_t = 1) &= \lambda dt \end{aligned}$$

Diffusion approximation example (2)

- We can compute explicitly the expectation and variance of the process increments, i.e.

$$\begin{aligned}
\mathrm{E}(dX_t) &= \sum_{dX_t} dX_t \cdot P(dX_t) \\
&= -1 \cdot x\mu dt + 0 \cdot (1 - (\lambda + x\mu)dt) + 1 \cdot \lambda dt \\
&= (\lambda - x\mu)dt
\end{aligned}$$

Diffusion approximation example (2)

- We can compute explicitly the expectation and variance of the process increments, i.e.

$$\begin{aligned}
\mathrm{E}(dX_t) &= \sum_{dX_t} dX_t \cdot P(dX_t) \\
&= -1 \cdot x\mu dt + 0 \cdot (1 - (\lambda + x\mu)dt) + 1 \cdot \lambda dt \\
&= (\lambda - x\mu)dt \\
\mathrm{Var}(dX_t) &= \sum_{dX_t} (dX_t - \mathrm{E}(dX_t))^2 \cdot P(dX_t) = \ldots = (\lambda + x\mu)dt
\end{aligned}$$

Diffusion approximation example (2)

- We can compute explicitly the expectation and variance of the process increments, i.e.

$$\begin{aligned}
\mathrm{E}(dX_t) &= \sum_{dX_t} dX_t \cdot P(dX_t) \\
&= -1 \cdot x\mu dt + 0 \cdot (1-(\lambda + x\mu)dt) + 1 \cdot \lambda dt \\
&= (\lambda - x\mu)dt \\
\mathrm{Var}(dX_t) &= \sum_{dX_t} (dX_t - \mathrm{E}(dX_t))^2 \cdot P(dX_t) = \ldots = (\lambda + x\mu)dt
\end{aligned}$$

- Using the approximation $\Delta X_t \sim N(\mathrm{E}(\Delta X_t), \mathrm{Var}(\Delta X_t))$, we obtain a diffusion approximation for the immigration-death process

$$dX_t = (\lambda - x\mu)dt + \sqrt{\lambda + x\mu}dB_t$$

Diffusion approximation example (3)

- A realization of the diffusion approximation of the immigration-death process

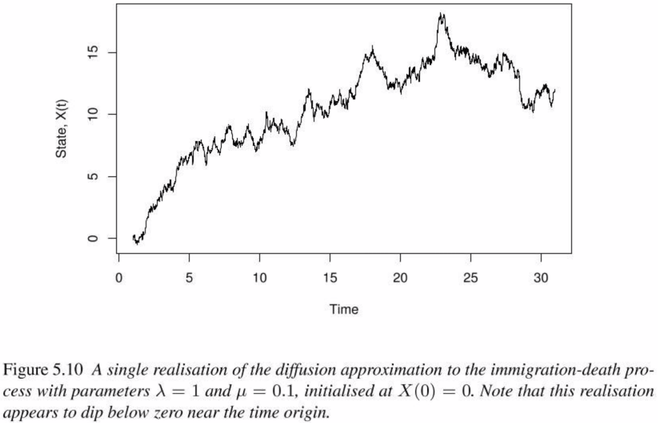

Figure 5.10 *A single realisation of the diffusion approximation to the immigration-death process with parameters $\lambda = 1$ and $\mu = 0.1$, initialised at $X(0) = 0$. Note that this realisation appears to dip below zero near the time origin.*

Figure: Figure 5.10 from (Wilkinson, 2011)

Wiener process

- A d-dimensional Brownian motion W (often called Wiener process) has d independent components, each of which is a univariate Brownian motion
- Thus,
  1. $W_0 = 0$ (zero-vector of length d)
  2. $W_t - W_s \sim N(0, (t-s) \cdot I_d)$, $\forall t > s$ (I_d is the identity matrix with size d)
  3. The increment $W_t - W_s$ is independent of the increment $W_{t'} - W_{s'}$, $\forall t > s \geq t' > s'$

Wiener process

- A d-dimensional Brownian motion W (often called Wiener process) has d independent components, each of which is a univariate Brownian motion
- Thus,
  1. $W_0 = 0$ (zero-vector of length d)
  2. $W_t - W_s \sim N(0, (t-s) \cdot I_d)$, $\forall t > s$ (I_d is the identity matrix with size d)
  3. The increment $W_t - W_s$ is independent of the increment $W_{t'} - W_{s'}$, $\forall t > s \geq t' > s'$
- As with the univariate Brownian motion, for small time increment Δt we can define a process increment

  $$\Delta W_t = W_{t+\Delta t} - W_t \sim N(0, \Delta t \cdot I_d)$$

  which again provides a simulation algorithm (with fixed time points/increments)

  $$W_0 = 0, \quad W_{\Delta t} = \Delta W^{(1)}, \quad W_{2\Delta t} = W_{\Delta t} + \Delta W^{(2)}, \quad W_{3\Delta t} = W_{2\Delta t} + \Delta W^{(3)}, \ldots$$

  where $\Delta W^{(i)}$ values denote realizations of the Gaussian-distributed process increment ΔW

Wiener process illustration

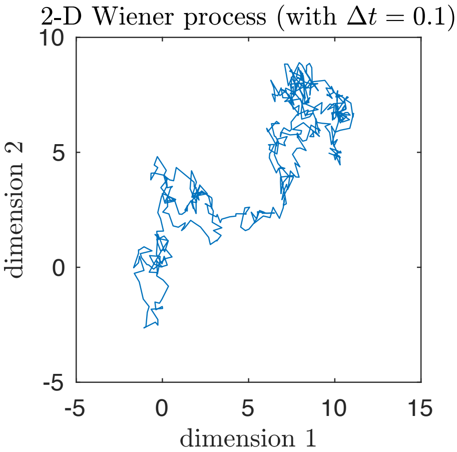

Figure: Wiener process illustration

Stochastic differential equation model

- A d-dimensional Itô diffusion process X_t is governed by a stochastic differential equation (SDE) model of the form

  $$dX_t = \mu(X_t)dt + \Psi(X_t)dW_t,$$

  where

  - $X_t \in \mathbb{R}^d$ is the state vector in continuous space
  - $\mu : \mathbb{R}^d \to \mathbb{R}^d$ is a deterministic drift function / vector
  - $\Psi : \mathbb{R}^d \to \mathbb{R}^d \times \mathbb{R}^d$ is a $(d \times d)$-dimensional diffusion matrix
  - W_t is a d-dimensional Wiener process

Stochastic differential equation model

- A d-dimensional Itô diffusion process X_t is governed by a stochastic differential equation (SDE) model of the form

  $$dX_t = \mu(X_t)dt + \Psi(X_t)dW_t,$$

  where
  - $X_t \in \mathbb{R}^d$ is the state vector in continuous space
  - $\mu \ : \ \mathbb{R}^d \to \mathbb{R}^d$ is a deterministic drift function / vector
  - $\Psi \ : \ \mathbb{R}^d \to \mathbb{R}^d \times \mathbb{R}^d$ is a $(d \times d)$-dimensional diffusion matrix
  - W_t is a d-dimensional Wiener process
- Loosely speaking, the SDE can be considered as a recipe for constructing a realization of X_t from a realization of a d-dimensional Wiener process

Stochastic differential equation model

- A d-dimensional Itô diffusion process X_t is governed by a stochastic differential equation (SDE) model of the form

  $$dX_t = \mu(X_t)dt + \Psi(X_t)dW_t,$$

  where

  - $X_t \in \mathbb{R}^d$ is the state vector in continuous space
  - $\mu : \mathbb{R}^d \to \mathbb{R}^d$ is a deterministic drift function / vector
  - $\Psi : \mathbb{R}^d \to \mathbb{R}^d \times \mathbb{R}^d$ is a $(d \times d)$-dimensional diffusion matrix
  - W_t is a d-dimensional Wiener process

- Loosely speaking, the SDE can be considered as a recipe for constructing a realization of X_t from a realization of a d-dimensional Wiener process
- The diffusion process approximation for the 1-D model discussed above generalizes directly to multivariate processes, i.e., given the value of X_t

  $$\Delta X_t = X_{t+\Delta t} - X_t \sim N(\mathrm{E}(\Delta X_t), \mathrm{Var}(X_t)),$$

  where

  $$\mathrm{E}(\Delta X_t) = \mu(X_t)\Delta t \quad \text{and} \quad \mathrm{Var}(X_t) = \Sigma(X_t)\Delta t = \Psi(X_t)\Psi(X_t)^T\Delta t$$

Euler-Maruyama algorithm

- If we define the increment in the diffusion process X_t using a small time increment Δt, then SDE can be interpreted as the limit (w.r.t. Δt) of the following difference equation

$$\Delta X_t = X_{t+\Delta t} - X_t = \mu(X_t)\Delta t + \Psi(X_t)\Delta W_t$$

Euler-Maruyama algorithm

- If we define the increment in the diffusion process X_t using a small time increment Δt, then SDE can be interpreted as the limit (w.r.t. Δt) of the following difference equation

$$\Delta X_t = X_{t+\Delta t} - X_t = \mu(X_t)\Delta t + \Psi(X_t)\Delta W_t$$

- For finite Δt this leads to the Euler-Maruyama algorithm for simulating SDEs

$$X_{t+\Delta t} = X_t + \mu(X_t)\Delta t + \Psi(X_t)\Delta W_t,$$

where again $\Delta W_t \sim N(0, \Delta t \cdot I_d)$

Euler-Maruyama algorithm

- If we define the increment in the diffusion process X_t using a small time increment Δt, then SDE can be interpreted as the limit (w.r.t. Δt) of the following difference equation

$$\Delta X_t = X_{t+\Delta t} - X_t = \mu(X_t)\Delta t + \Psi(X_t)\Delta W_t$$

- For finite Δt this leads to the Euler-Maruyama algorithm for simulating SDEs

$$X_{t+\Delta t} = X_t + \mu(X_t)\Delta t + \Psi(X_t)\Delta W_t,$$

  where again $\Delta W_t \sim N(0, \Delta t \cdot I_d)$

- The above equation can be applied recursively (from initial value X_{init} at time $t = 0$ or other initial time) to obtain a value of the process for time points $0, \Delta t, 2\Delta t, \ldots$

$$\begin{aligned} X_0 &= X_{\text{init}} \\ X_{\Delta t} &= X_0 + \mu(X_0)\Delta t + \Psi(X_0)\Delta W^{(0)} \\ X_{2\Delta t} &= X_{\Delta t} + \mu(X_{\Delta t})\Delta t + \Psi(X_{\Delta t})\Delta W^{(1)}, \ldots \end{aligned}$$

  where $\Delta W^{(i)}$ values are realizations of $N(0, \Delta t \cdot I_d)$

Illustration of Euler-Maruyama

- Consider dynamics of a stochastic variant of a so-called Van der Pol system defined as the following SDE

$$dX_t = \mu(X_t)dt + \Psi(X_t)dW_t,$$

where

- $X_t = (X_{t1}, X_{t2})^T \in \mathbb{R}^2$ is the state vector in continuous space
- $\mu \ : \ \mathbb{R}^2 \rightarrow \mathbb{R}^2$ is a deterministic drift function / vector

$$\mu\left(\begin{bmatrix} X_{t1} \\ X_{t2} \end{bmatrix}\right) = \begin{bmatrix} X_{t2} \\ (1 - X_{t1})^2 X_{t2} - X_{t1} \end{bmatrix}$$

- $\Psi \ : \ \mathbb{R}^d \rightarrow \mathbb{R}^d \times \mathbb{R}^d$ is a $(d \times d)$-dimensional diffusion matrix

$$\Psi(X_t) = \sigma^2 \cdot I_2 \ \text{ or } \ \Psi(X_t) = N(X_t|(-2,0)^T, I_2)$$

  (in the latter the diffusion function is just the value of the normal density at X_t, not a random variable)
- W_t is a d-dimensional Wiener process

Illustration of Euler-Maruyama (2)

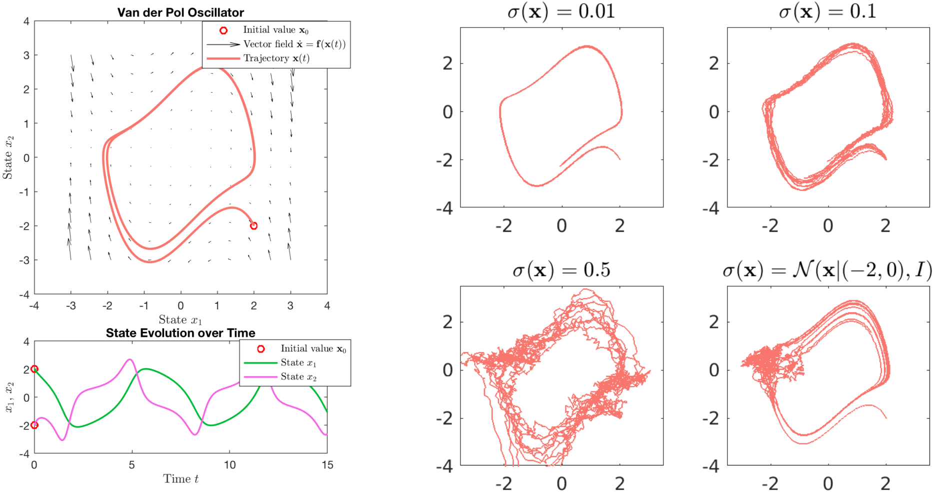

Figure: Dynamics of stochastic Van der Pol simulated by the Euler-Maruyama (Yildiz et al., 2018)

Chemical Langevin equation

- Motivation: use diffusion approximation of the true process to accelerate simulation of biochemical reaction networks
- Recall:
  - For coupled chemical reaction networks: $X^* = X + Sr$ or $\Delta X = Sr$
  - The Poisson timestep (approximative) simulation method

Chemical Langevin equation

- Motivation: use diffusion approximation of the true process to accelerate simulation of biochemical reaction networks
- Recall:
  - For coupled chemical reaction networks: $X^* = X + Sr$ or $\Delta X = Sr$
  - The Poisson timestep (approximative) simulation method
- In an infinitesimal interval dt the change in state, dXt, is SdR_t, where dR_t is a v-vector whose ith element is a random variable with density

$$\mathrm{Po}(h_i(X_t, c_i)dt)$$

- Recall that for a random variable $Z \sim \mathrm{Po}(\lambda)$ mean and variance are:

$$\mathrm{E}(Z) = \mathrm{Var}(Z) = \lambda$$

Chemical Langevin equation

- Motivation: use diffusion approximation of the true process to accelerate simulation of biochemical reaction networks
- Recall:
  - For coupled chemical reaction networks: $X^* = X + Sr$ or $\Delta X = Sr$
  - The Poisson timestep (approximative) simulation method
- In an infinitesimal interval dt the change in state, dXt, is SdR_t, where dR_t is a v-vector whose ith element is a random variable with density

$$\mathrm{Po}(h_i(X_t, c_i)dt)$$

- Recall that for a random variable $Z \sim \mathrm{Po}(\lambda)$ mean and variance are:

$$\mathrm{E}(Z) = \mathrm{Var}(Z) = \lambda$$

- Thus by matching the first two moments, mean and variance, we obtain

$$dR_t = h(X_t, c)dt + \mathrm{diag}\{\sqrt{h(X_t, c)}\}dW_t,$$

where $h(X, c) = (h_1(X, c_1), \ldots, h_v(X, c_v))^T$

Chemical Langevin equation (2)

- We now obtain the diffusion approximation

$$\begin{aligned} dX_t &= SdR_t \\ &= S\left(h(X_t, c)dt + \mathrm{diag}\{\sqrt{h(X_t, c)}\}dW_t\right) \\ &= Sh(X_t, c)dt + S\mathrm{diag}\{\sqrt{h(X_t, c)}\}dW_t \end{aligned}$$

- This is called the chemical Langevin equation

Chemical Langevin equation (2)

- We now obtain the diffusion approximation

$$\begin{aligned} dX_t &= SdR_t \\ &= S\left(h(X_t, c)dt + \mathrm{diag}\{\sqrt{h(X_t, c)}\}dW_t\right) \\ &= Sh(X_t, c)dt + S\mathrm{diag}\{\sqrt{h(X_t, c)}\}dW_t \end{aligned}$$

- This is called the chemical Langevin equation
- Notice that
  - $X_t \in \mathbb{R}^u$, where u denotes again the number of species
  - S is the stoichiometric matrix that has size u-by-v, where v denotes the number of reactions
  - $h(X_t, c) \in \mathbb{R}_+^v$ is a vector that has length v
  - $\mathrm{diag}\{\sqrt{h(X_t, c)}\}$ is a matrix that has size v-by-v
  - $dW_t \in \mathbb{R}^v$, i.e., a vector that has as many elements as there are reactions

Chemical Langevin equation (2)

- We now obtain the diffusion approximation

$$\begin{aligned} dX_t &= SdR_t \\ &= S\left(h(X_t, c)dt + \mathrm{diag}\{\sqrt{h(X_t, c)}\}dW_t\right) \\ &= Sh(X_t, c)dt + S\mathrm{diag}\{\sqrt{h(X_t, c)}\}dW_t \end{aligned}$$

- This is called the chemical Langevin equation
- Notice that
  - $X_t \in \mathbb{R}^u$, where u denotes again the number of species
  - S is the stoichiometric matrix that has size u-by-v, where v denotes the number of reactions
  - $h(X_t, c) \in \mathbb{R}^v_+$ is a vector that has length v
  - $\mathrm{diag}\{\sqrt{h(X_t, c)}\}$ is a matrix that has size v-by-v
  - $dW_t \in \mathbb{R}^v$, i.e., a vector that has as many elements as there are reactions
- Notice that if there is no diffusion (Wiener process), then we retrieve the continuous deterministic model

$$dX_t = Sh(X_t, c)dt$$

References

- Darren J. Wilkinson, *Stochastic Modelling for Systems Biology*, Chapman & Hall/CRC, 2011
- Yildiz C, Heinonen M, Mannerström H, Intosalmi J, and Lähdesmäki H, Learning stochastic differential equations with Gaussian processes without gradient matching, In *IEEE International Workshop on Machine Learning for Signal Processing*, 2018.