9

Approximation of Nonlinear SDEs

This chapter is concerned with approximations of nonlinear SDEs aiming for computing statistics of SDEs and for forming approximations to the transition densities. However, many of them can also be used for numerical simulation of SDEs. These approximations will also turn out to be useful in filtering and smoothing theory as well as in SDE parameter estimation, which we consider in the subsequent chapters. In this chapter, we start with Gaussian approximations, which are especially common in filtering and smoothing applications (such as target tracking). These approximations include various linearization, Taylor series, and sigma-point approximations. We then proceed to temporal Taylor series approximations of the moments, Hermite expansions of the transition densities, discretization approximations to the Fokker–Planck–Kolmogorov equation, and finally pathwise series expansions of Brownian motions, which are related to the Wong–Zakai theorem and Wiener chaos expansions.

9.1 Gaussian Assumed Density Approximations

In Section 5.5, we saw that the differential equations for the mean and covariance of the solution to the SDE

$$\mathrm{d}\mathbf{x} = \mathbf{f}(\mathbf{x}, t)\,\mathrm{d}t + \mathbf{L}(\mathbf{x}, t)\,\mathrm{d}\boldsymbol{\beta} \tag{9.1}$$

are

$$\frac{\mathrm{d}\mathbf{m}}{\mathrm{d}t} = \mathrm{E}\left[\mathbf{f}(\mathbf{x}, t)\right], \tag{9.2}$$

$$\begin{aligned}\frac{\mathrm{d}\mathbf{P}}{\mathrm{d}t} &= \mathrm{E}\left[\mathbf{f}(\mathbf{x}, t)\,(\mathbf{x} - \mathbf{m})^{\mathsf{T}}\right] + \mathrm{E}\left[(\mathbf{x} - \mathbf{m})\,\mathbf{f}^{\mathsf{T}}(\mathbf{x}, t)\right] \\ &\quad + \mathrm{E}\left[\mathbf{L}(\mathbf{x}, t)\,\mathbf{Q}\,\mathbf{L}^{\mathsf{T}}(\mathbf{x}, t)\right].\end{aligned} \tag{9.3}$$

If we write down the expectation integrals explicitly, these equations can be seen to have the form

$$\frac{\mathrm{d}\mathbf{m}}{\mathrm{d}t} = \int \mathbf{f}(\mathbf{x},t)\, p(\mathbf{x},t)\, \mathrm{d}\mathbf{x}, \tag{9.4}$$

$$\begin{aligned} \frac{\mathrm{d}\mathbf{P}}{\mathrm{d}t} &= \int \mathbf{f}(\mathbf{x},t)\, (\mathbf{x}-\mathbf{m})^{\mathsf{T}}\, p(\mathbf{x},t)\, \mathrm{d}\mathbf{x} \\ &\quad + \int (\mathbf{x}-\mathbf{m})\, \mathbf{f}^{\mathsf{T}}(\mathbf{x},t)\, p(\mathbf{x},t)\, \mathrm{d}\mathbf{x} \\ &\quad + \int \mathbf{L}(\mathbf{x},t)\, \mathbf{Q}\, \mathbf{L}^{\mathsf{T}}(\mathbf{x},t)\, p(\mathbf{x},t)\, \mathrm{d}\mathbf{x}. \end{aligned} \tag{9.5}$$

Because $p(\mathbf{x},t)$ is the solution of the Fokker–Planck–Kolmogorov equation (5.11), these equations cannot usually be solved in practice. However, one very useful class of approximations can be obtained by replacing the FPK solution with a Gaussian approximation as follows:

$$p(\mathbf{x},t) \approx \mathrm{N}(\mathbf{x} \mid \mathbf{m}(t), \mathbf{P}(t)), \tag{9.6}$$

where $\mathbf{m}(t)$ and $\mathbf{P}(t)$ are the mean and covariance of the state, respectively. This approximation is referred to as the *Gaussian assumed density approximation* (Kushner, 1967; Särkkä and Sarmavuori, 2013), because we do the computations under the assumption that the state distribution is indeed Gaussian. It is also related to a *Gaussian process approximation* (Archambeau and Opper, 2011; Ala-Luhtala et al., 2015) of the SDE. The approximation method (as it is presented in Särkkä and Sarmavuori, 2013) can be written as the following algorithm.

Algorithm 9.1 (Gaussian assumed density approximation I)**.** *A Gaussian process approximation to the SDE* (9.1) *can be obtained by integrating the following differential equations from the initial conditions* $\mathbf{m}(t_0) = \mathrm{E}[\mathbf{x}(t_0)]$ *and* $\mathbf{P}(t_0) = \mathrm{Cov}[\mathbf{x}(t_0)]$ *to the target time* t*:*

$$\begin{aligned} \frac{\mathrm{d}\mathbf{m}}{\mathrm{d}t} &= \int \mathbf{f}(\mathbf{x},t)\ \mathrm{N}(\mathbf{x} \mid \mathbf{m}, \mathbf{P})\, \mathrm{d}\mathbf{x}, \\ \frac{\mathrm{d}\mathbf{P}}{\mathrm{d}t} &= \int \mathbf{f}(\mathbf{x},t)\, (\mathbf{x}-\mathbf{m})^{\mathsf{T}}\ \mathrm{N}(\mathbf{x} \mid \mathbf{m}, \mathbf{P})\, \mathrm{d}\mathbf{x} \\ &\quad + \int (\mathbf{x}-\mathbf{m})\, \mathbf{f}^{\mathsf{T}}(\mathbf{x},t)\ \mathrm{N}(\mathbf{x} \mid \mathbf{m}, \mathbf{P})\, \mathrm{d}\mathbf{x} \\ &\quad + \int \mathbf{L}(\mathbf{x},t)\, \mathbf{Q}\, \mathbf{L}^{\mathsf{T}}(\mathbf{x},t)\ \mathrm{N}(\mathbf{x} \mid \mathbf{m}, \mathbf{P})\, \mathrm{d}\mathbf{x}. \end{aligned} \tag{9.7}$$

If we denote the Gaussian expectation as

$$\mathrm{E}_{\mathrm{N}}[\mathbf{g}(\mathbf{x})] = \int \mathbf{g}(\mathbf{x}) \; \mathrm{N}(\mathbf{x} \mid \mathbf{m}, \mathbf{P}) \, \mathrm{d}\mathbf{x}, \tag{9.8}$$

then the equations can also be written as

$$\begin{aligned} \frac{\mathrm{d}\mathbf{m}}{\mathrm{d}t} &= \mathrm{E}_{\mathrm{N}}[\mathbf{f}(\mathbf{x}, t)], \\ \frac{\mathrm{d}\mathbf{P}}{\mathrm{d}t} &= \mathrm{E}_{\mathrm{N}}[(\mathbf{x} - \mathbf{m}) \, \mathbf{f}^{\mathsf{T}}(\mathbf{x}, t)] + \mathrm{E}_{\mathrm{N}}[\mathbf{f}(\mathbf{x}, t) \, (\mathbf{x} - \mathbf{m})^{\mathsf{T}}] \\ &\quad + \mathrm{E}_{\mathrm{N}}[\mathbf{L}(\mathbf{x}, t) \, \mathbf{Q} \, \mathbf{L}^{\mathsf{T}}(\mathbf{x}, t)]. \end{aligned} \tag{9.9}$$

If the function $\mathbf{x} \mapsto \mathbf{f}(\mathbf{x}, t)$ is differentiable, the covariance differential equation can be simplified by using the following well-known property of Gaussian random variables (also known as Stein's lemma).

Theorem 9.2. *Let* $\mathbf{f}(\mathbf{x}, t)$ *be differentiable with respect to* $\mathbf{x}$ *and let* $\mathbf{x} \sim \mathrm{N}(\mathbf{m}, \mathbf{P})$. *Then the following identity holds (see, e.g., Papoulis, 1984; Särkkä and Sarmavuori, 2013):*

$$\begin{aligned} &\int \mathbf{f}(\mathbf{x}, t) \, (\mathbf{x} - \mathbf{m})^{\mathsf{T}} \; \mathrm{N}(\mathbf{x} \mid \mathbf{m}, \mathbf{P}) \, \mathrm{d}\mathbf{x} \\ &\qquad = \left[\int \mathbf{F}_{\mathbf{x}}(\mathbf{x}, t) \; \mathrm{N}(\mathbf{x} \mid \mathbf{m}, \mathbf{P}) \, \mathrm{d}\mathbf{x} \right] \mathbf{P}, \end{aligned} \tag{9.10}$$

where $\mathbf{F}_{\mathbf{x}}(\mathbf{x}, t)$ *is the Jacobian matrix of* $\mathbf{f}(\mathbf{x}, t)$ *with respect to* $\mathbf{x}$.

Using the theorem, the mean and covariance Equations (9.9) can be equivalently written as follows.

Algorithm 9.3 (Gaussian assumed density approximation II)**.** *A Gaussian process approximation to the SDE* (9.1) *can be obtained by integrating the following differential equations from the initial conditions* $\mathbf{m}(t_0) = \mathrm{E}[\mathbf{x}(t_0)]$ *and* $\mathbf{P}(t_0) = \mathrm{Cov}[\mathbf{x}(t_0)]$ *to the target time* t*:*

$$\begin{aligned} \frac{\mathrm{d}\mathbf{m}}{\mathrm{d}t} &= \mathrm{E}_{\mathrm{N}}[\mathbf{f}(\mathbf{x}, t)], \\ \frac{\mathrm{d}\mathbf{P}}{\mathrm{d}t} &= \mathbf{P} \; \mathrm{E}_{\mathrm{N}}[\mathbf{F}_{\mathbf{x}}(\mathbf{x}, t)]^{\mathsf{T}} + \mathrm{E}_{\mathrm{N}}[\mathbf{F}_{\mathbf{x}}(\mathbf{x}, t)] \, \mathbf{P} + \mathrm{E}_{\mathrm{N}}[\mathbf{L}(\mathbf{x}, t) \, \mathbf{Q} \, \mathbf{L}^{\mathsf{T}}(\mathbf{x}, t)], \end{aligned} \tag{9.11}$$

where $\mathrm{E}_{\mathrm{N}}[\bullet]$ *denotes the expectation with respect to* $\mathbf{x} \sim \mathrm{N}(\mathbf{m}, \mathbf{P})$.

The approximations presented in this section are formally equivalent to so-called statistical linearization approximations (Gelb, 1974; Socha, 2008), and they are also closely related to the variational approximations of Archambeau and Opper (2011).

Although the preceding algorithms provide a generic Gaussian assumed density approximation framework for SDEs, to implement the methods, one is required to compute the following kind of n-dimensional Gaussian integrals:

$$\mathrm{E}_{\mathrm{N}}[\mathbf{g}(\mathbf{x}, t)] = \int \mathbf{g}(\mathbf{x}, t) \; \mathrm{N}(\mathbf{x} \mid \mathbf{m}, \mathbf{P}) \, \mathrm{d}\mathbf{x}. \tag{9.12}$$

A classical approach that is very common in filtering theory (Jazwinski, 1970; Maybeck, 1982a) is to linearize (via Taylor series) the drift $\mathbf{f}(\mathbf{x}, t)$ around the mean $\mathbf{m}$ as

$$\mathbf{f}(\mathbf{x}, t) \approx \mathbf{f}(\mathbf{m}, t) + \mathbf{F}_{\mathbf{x}}(\mathbf{m}, t) \, (\mathbf{x} - \mathbf{m}) \tag{9.13}$$

and to approximate the expectation of the diffusion part as

$$\mathbf{L}(\mathbf{x}, t) \approx \mathbf{L}(\mathbf{m}, t). \tag{9.14}$$

This leads to the following approximation, which is commonly used in extended Kalman filters (EKF).

Algorithm 9.4 (Linearization approximation of SDE). *A linearization-based approximation to the SDE* (9.1) *can be obtained by integrating the following differential equations from the initial conditions* $\mathbf{m}(t_0) = \mathrm{E}[\mathbf{x}(t_0)]$ *and* $\mathbf{P}(t_0) = \mathrm{Cov}[\mathbf{x}(t_0)]$ *to the target time* t*:*

$$\begin{aligned} \frac{\mathrm{d}\mathbf{m}}{\mathrm{d}t} &= \mathbf{f}(\mathbf{m}, t), \\ \frac{\mathrm{d}\mathbf{P}}{\mathrm{d}t} &= \mathbf{P} \, \mathbf{F}_{\mathbf{x}}^{\mathsf{T}}(\mathbf{m}, t) + \mathbf{F}_{\mathbf{x}}(\mathbf{m}, t) \, \mathbf{P} \, + \mathbf{L}(\mathbf{m}, t) \, \mathbf{Q} \, \mathbf{L}^{\mathsf{T}}(\mathbf{m}, t). \end{aligned} \tag{9.15}$$

Another general class of approximations is the Gauss–Hermite cubature type of approximations, where the integrals are approximated as weighted sums

$$\int \mathbf{f}(\mathbf{x}, t) \; \mathrm{N}(\mathbf{x} \mid \mathbf{m}, \mathbf{P}) \, \mathrm{d}\mathbf{x} \approx \sum_i W^{(i)} \, \mathbf{f}(\mathbf{x}^{(i)}, t), \tag{9.16}$$

where $\mathbf{x}^{(i)}$ and $W^{(i)}$ are the sigma points (abscissas) and their accompanying weights, which have been selected using a method-specific deterministic rule. These kinds of rules are nowadays commonly used in the context of filtering theory (cf. Särkkä and Sarmavuori, 2013; Särkkä, 2013). In n-dimensional Gauss–Hermite integration, the unscented transform, and cubature integration, the sigma points are selected as follows:

$$\mathbf{x}^{(i)} = \mathbf{m} + \sqrt{\mathbf{P}}\,\boldsymbol{\xi}_i, \tag{9.17}$$

where the matrix square root is defined by $\mathbf{P} = \sqrt{\mathbf{P}}\,\sqrt{\mathbf{P}}^{\mathsf{T}}$ (typically Cholesky factorization), and the points $\boldsymbol{\xi}_i$ and the weights $W^{(i)}$ are selected as follows:

Gauss–Hermite integration: This product rule–based method uses a set of m^n vectors $\boldsymbol{\xi}_i$, which have been formed as a Cartesian product of zeros of the Hermite polynomials of order m. The weights $W^{(i)}$ are formed as products of the corresponding one-dimensional Gauss–Hermite integration weights (for details, see Ito and Xiong, 2000; Wu et al., 2006).

Unscented transform: This method uses a zero vector (origin) and $2n$ unit coordinate vectors $\mathbf{e}_i$ as follows (the method can also be generalized a bit):

$$\begin{aligned} \boldsymbol{\xi}_0 &= \mathbf{0}, \\ \boldsymbol{\xi}_i &= \begin{cases} \sqrt{\lambda + n}\,\mathbf{e}_i, & i = 1, 2, \ldots, n, \\ -\sqrt{\lambda + n}\,\mathbf{e}_{i-n}, & i = n+1, n+2, \ldots, 2n, \end{cases} \end{aligned} \tag{9.18}$$

and the weights are defined as follows:

$$\begin{aligned} W^{(0)} &= \frac{\lambda}{n + \kappa}, \\ W^{(i)} &= \frac{1}{2(n+\kappa)}, \qquad i = 1, 2, \ldots, 2n, \end{aligned} \tag{9.19}$$

where κ and λ are parameters of the method and n is the dimensionality of $\mathbf{x}$ (see, e.g., Julier et al., 1995; Wan and van der Merwe, 2001; Julier and Uhlmann, 2004; Särkkä, 2013). This rule is a special case of more general symmetric integrations rules that can be constructed up to an arbitrary order (McNamee and Stenger, 1967).

Cubature method: This spherical third-degree method uses only $2n$ vectors as follows:

$$\boldsymbol{\xi}_i = \begin{cases} \sqrt{n}\,\mathbf{e}_i, & i = 1, 2, \ldots, n, \\ -\sqrt{n}\,\mathbf{e}_{i-n}, & i = n+1, n+2, \ldots, 2n, \end{cases} \tag{9.20}$$

and the weights are defined as $W^{(i)} = 1/(2n)$, for $i = 1, 2, \ldots, 2n$. This rule is in fact a special case of the preceding rule, but has turned out to be useful in filtering theory (Arasaratnam and Haykin, 2009; Arasaratnam et al., 2010; Särkkä and Solin, 2012; Särkkä, 2013).

The sigma-point methods lead to the following approximations to the mean and covariance differential equations.

Algorithm 9.5 (Sigma-point approximation of SDEs)**.** *A sigma point–based approximation to the SDE* (9.1) *can be obtained by integrating the following differential equations from the initial conditions* $\mathbf{m}(t_0) = \mathrm{E}[\mathbf{x}(t_0)]$ *and* $\mathbf{P}(t_0) = \mathrm{Cov}[\mathbf{x}(t_0)]$ *to the target time* t*:*

$$\begin{aligned}
\frac{\mathrm{d}\mathbf{m}}{\mathrm{d}t} &= \sum_i W^{(i)}\,\mathbf{f}(\mathbf{m} + \sqrt{\mathbf{P}}\,\boldsymbol{\xi}_i, t), \\
\frac{\mathrm{d}\mathbf{P}}{\mathrm{d}t} &= \sum_i W^{(i)}\,\mathbf{f}(\mathbf{m} + \sqrt{\mathbf{P}}\,\boldsymbol{\xi}_i, t)\,\boldsymbol{\xi}_i^{\mathsf{T}}\,\sqrt{\mathbf{P}}^{\mathsf{T}} \\
&\quad + \sum_i W^{(i)}\,\sqrt{\mathbf{P}}\,\boldsymbol{\xi}_i\,\mathbf{f}^{\mathsf{T}}(\mathbf{m} + \sqrt{\mathbf{P}}\,\boldsymbol{\xi}_i, t) \\
&\quad + \sum_i W^{(i)}\,\mathbf{L}(\mathbf{m} + \sqrt{\mathbf{P}}\,\boldsymbol{\xi}_i, t)\,\mathbf{Q}\,\mathbf{L}^{\mathsf{T}}(\mathbf{m} + \sqrt{\mathbf{P}}\,\boldsymbol{\xi}_i, t).
\end{aligned} \tag{9.21}$$

Once the Gaussian integral approximation has been selected, the solutions to the resulting ordinary differential equations can be computed, for example, by the fourth-order Runge–Kutta method or some similar numerical ODE solution method. It would also be possible to approximate the integrals using various other methods from filtering theory (see, e.g., Jazwinski, 1970; Wu et al., 2006; Särkkä and Sarmavuori, 2013).

Example 9.6 (Linearization and Gauss–Hermite approximations)**.** *The linearization-based and sigma-point approximations of the Beneš SDE*

$$\mathrm{d}x = \tanh(x)\,\mathrm{d}t + \mathrm{d}\beta, \quad x(0) = x_0, \tag{9.22}$$

where $\beta(t)$ *is a standard Brownian motion, which we already considered in Example 7.10, look like the following:*

- *The linearization approximation is*

$$\begin{aligned}
\frac{\mathrm{d}m}{\mathrm{d}t} &= \tanh(m), \\
\frac{\mathrm{d}P}{\mathrm{d}t} &= 2(1 - \tanh^2(m))\,P + 1.
\end{aligned} \tag{9.23}$$

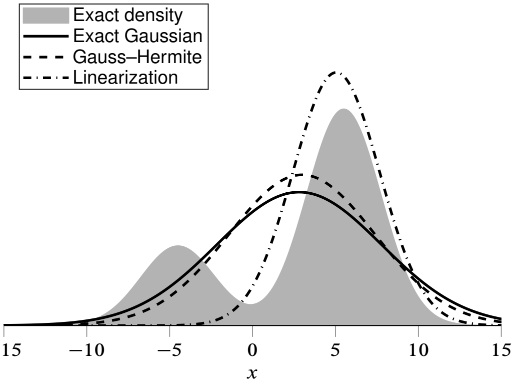


Figure 9.1 Comparison of the Gaussian approximations in Example 9.6 to the exact density and to the exact Gaussian fit.

- *A generic sigma-point approximation is given as*

$$\begin{aligned}\frac{\mathrm{d}m}{\mathrm{d}t} &= \sum_i W^{(i)} \tanh(m + \sqrt{P}\,\xi_i),\\ \frac{\mathrm{d}P}{\mathrm{d}t} &= 1 + 2\sum_i W^{(i)} \sqrt{P}\,\xi_i \tanh(m + \sqrt{P}\,\xi_i).\end{aligned} \tag{9.24}$$

The comparison of these approximations to the exact solutions is shown in Figure 9.1. The approximations are compared to the exact density and to the exact Gaussian fit formed by moment-matching to the exact density. The starting point was $x(0) = 1/2$ and the solution was computed at time $t = 5$. The sigma-point method was a 10-point Gauss–Hermite quadrature approximation. As can be seen in the figure, the sigma-point method produces a more accurate match to the moment-matched Gaussian than the linearization approximation. However, it is apparent that a Gaussian approximation poorly approximates the true multi-modal distribution.

Example 9.7 (Gaussian approximation of a nonlinear trigonometric SDE)**.** *Consider the nonlinear Itô stochastic differential equation model that was already covered in Example 8.6:*

$$\mathrm{d}x = -\left(\frac{1}{10}\right)^2 \sin(x)\,\cos^3(x)\,\mathrm{d}t + \frac{1}{10}\cos^2(x)\,\mathrm{d}\beta, \quad x(0) = x_0, \tag{9.25}$$

where $\beta(t)$ is a standard Brownian motion. The SDE has the solution

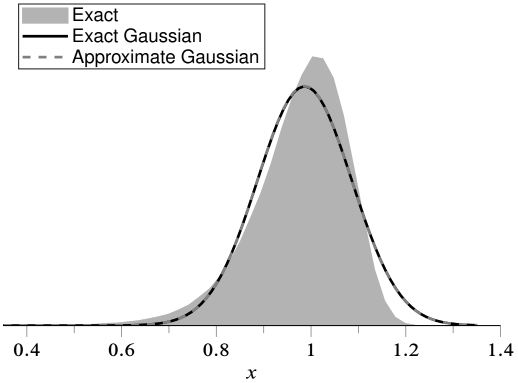


Figure 9.2 An example of a sigma point–based Gaussian approximation to a nonlinear SDE in Example 9.7. The exact solution at $t = 10$ is shown by the patch, and the solid line illustrates the Gaussian fit to it. The approximation is shown by the dashed line.

$x(t) = \arctan(1/10\ \beta(t) + \tan(x_0))$. *In this example, let* $x_0 = 1$. *The goal is to characterize the solution at* $t = 10$ *using a Gaussian approximation* $p(x(t)) \approx \mathrm{N}(x(t) \mid m(t), P(t))$ *of the exact solution.*

In the model, we have the drift $f(x) = -(1/10)^2 \sin(x) \cos^3(x)$ *and dispersion coefficient* $L(x) = 1/10 \cos^2(x)$. *Applying the cubature integration sigma-point scheme (*$\xi = \pm 1, W^{(i)} = 1/2\ , i = 1, 2$*) gives the following mean and covariance differential equations:*

$$\frac{\mathrm{d}m}{\mathrm{d}t} = \frac{1}{2} f(m - \sqrt{P}) + \frac{1}{2} f(m + \sqrt{P}), \tag{9.26}$$

$$\begin{aligned} \frac{\mathrm{d}P}{\mathrm{d}t} &= \sqrt{P}\, f(m - \sqrt{P}) - \sqrt{P}\, f(m + \sqrt{P}) \\ &\quad + \frac{1}{2} \left[L(m - \sqrt{P}) \right]^2 + \frac{1}{2} \left[L(m + \sqrt{P}) \right]^2, \end{aligned} \tag{9.27}$$

where $m(0) = x_0$, $P(0) = 0$. *We use the fourth-order Runge–Kutta scheme for solving* $m(10)$ *and* $P(10)$ *with a step size of* $\Delta t = 2^{-6}$. *Figure 9.2 illustrates the exact solution of* $x(10)$ *and shows the moment-matched Gaussian fit to it by a solid line. The ODE-based approximative solution* $\mathrm{N}(x(10) \mid m(10), P(10))$ *is shown by a dashed line, and it coincides well with the Gaussian fit.*

It is worth noting that all the Gaussian assumed density approximations, including the linearization and sigma-point methods discussed in this section (Algorithms 9.1–9.5), also provide Gaussian approximations for the transition densities.

Algorithm 9.8 (Gaussian assumed density transition density approximation)**.** *The Gaussian assumed density approximation for transition density* $p(\mathbf{x}(t) \mid \mathbf{x}(s))$ *where* $t > s$ *can be constructed by starting from the initial conditions* $\mathbf{m}(s) = \mathbf{x}(s)$*,* $\mathbf{P}(s) = \mathbf{0}$ *in any of the Algorithms 9.1–9.5 and solving the equations at time* t*. If we denote the resulting mean and covariance as* $\mathbf{m}(t \mid s)$ *and* $\mathbf{P}(t \mid s)$*, respectively, then the transition density approximation is given as*

$$p(\mathbf{x}(t) \mid \mathbf{x}(s)) \approx \mathrm{N}(\mathbf{x}(t) \mid \mathbf{m}(t \mid s), \mathbf{P}(t \mid s)). \tag{9.28}$$

Because all the Gaussian assumed density approximations produce a local approximation to the transition density, they can also be used as SDE simulation methods. The basic idea is to fix a small Δt and sequentially simulate transitions over Δt intervals by using the transition density approximation. This can be written as the following algorithm.

Algorithm 9.9 (SDE simulation by Gaussian assumed density approximation)**.** *Gaussian assumed density approximation–based numerical simulation of an SDE solution over an interval* $[t_0, t]$ *can be performed as follows:*

- *Divide the interval to* M *subintervals of length* Δt*. These interval lengths can also vary if required.*
- *Starting from* $\hat{\mathbf{x}}(t_0) \sim p(\mathbf{x}(t_0))$*, sequentially at each of the* M *subintervals* k *perform a draw from the approximate transition density as follows:*

  $$\hat{\mathbf{x}}(t_{k+1}) \sim \mathrm{N}(\mathbf{m}(t_{k+1} \mid t_k), \mathbf{P}(t_{k+1} \mid t_k)), \tag{9.29}$$

  where the transition density approximation of the right is computed with Algorithm 9.8 with $\hat{\mathbf{x}}(t_k)$ *as the starting point.*

Algorithm 9.9 has the property that unlike, for example, the Euler–Maruyama method, it is exact for linear systems. The convergence properties are still to be investigated, but they could be expected to be similar to the methods of Ozaki (1992, 1993) and Shoji and Ozaki (1998). Numerical comparison of the Gaussian assumed density approximation–based simulation method to local linearization is provided in Example 9.14.

The Gaussian assumed density approximations can be generalized to the exponential family of distributions by considering the projections of SDEs on manifolds of exponential densities (Brigo et al., 1999). The linearization approximations can also be formed along a nominal trajectory instead of the approximated mean trajectory (Maybeck, 1982a). Yet another possibility is to use a sigma-point version of the nominal trajectory methodology (García-Fernández et al., 2015, 2017).

9.2 Linearized Discretizations

One way to approximate the solutions of SDEs using Gaussian distributions is to first use methods such as Itô–Taylor series expansions or stochastic Runge–Kutta methods for forming a discrete-time approximation to the SDE, and then approximate it using Gaussian assumed density approximations. That is, if we originally have an SDE

$$\mathrm{d}\mathbf{x} = \mathbf{f}(\mathbf{x}, t)\,\mathrm{d}t + \mathbf{L}(\mathbf{x}, t)\,\mathrm{d}\boldsymbol{\beta}, \tag{9.30}$$

we then form an approximation as

$$\mathbf{x}(t_{k+1}) = \mathbf{f}_k^{\Delta t}(\mathbf{x}(t_k), \mathbf{e}_k), \tag{9.31}$$

where $\mathbf{f}_k^{\Delta t}$ is defined by the discretization method and $\mathbf{e}_k$ is a noise process related to the discretization. We can then step over $t_1, t_2, \ldots$ such that we always compute the mean and covariance of the next SDE state $\mathbf{x}(t_{k+1})$ assuming that the previous state $\mathbf{x}(t_k)$ was Gaussian with the mean and covariance that we calculated for it.

As we assume Gaussianity, the integrals to be recursively computed for $k = 1, 2, \ldots$ are the following:

$$\mathbf{m}_{k+1} = \iint \mathbf{f}_k^{\Delta t}(\mathbf{x}(t_k), \mathbf{e}_k)\ \mathrm{N}(\mathbf{x}(t_k) \mid \mathbf{m}_k, \mathbf{P}_k)\, p(\mathbf{e}_k)\,\mathrm{d}\mathbf{x}(t_k)\,\mathrm{d}\mathbf{e}_k, \tag{9.32}$$

$$\begin{aligned}\mathbf{P}_{k+1} = \iint & \left[\mathbf{f}_k^{\Delta t}(\mathbf{x}(t_k), \mathbf{e}_k) - \mathbf{m}_{k+1}\right]\left[\mathbf{f}_k^{\Delta t}(\mathbf{x}(t_k), \mathbf{e}_k) - \mathbf{m}_{k+1}\right]^{\mathsf{T}} \\ & \times \mathrm{N}(\mathbf{x}(t_k) \mid \mathbf{m}_k, \mathbf{P}_k)\, p(\mathbf{e}_k)\,\mathrm{d}\mathbf{x}(t_k)\,\mathrm{d}\mathbf{e}_k.\end{aligned} \tag{9.33}$$

If the noises $\mathbf{e}_k$ are Gaussian as well, then the joint distribution of $(\mathbf{x}(t_k), \mathbf{e}_k)$ is Gaussian and we can use the numerical integration methods outlined in the previous section, because the integrals have the Gaussian form (9.12). Because $\mathbf{e}_k$ is independent of $(\mathbf{x}(t_k), \mathbf{e}_k)$, we can also often integrate the noise out in closed form.

Example 9.10 (Approximative Itô–Taylor method)**.** *In the Itô–Taylor* *method in Algorithm 8.4, we have*

$$\begin{aligned}\mathbf{f}_k^{\Delta t}(\mathbf{x}(t_k), \mathbf{e}_k) &= \hat{\mathbf{x}}(t_k) + \mathbf{f}(\hat{\mathbf{x}}(t_k), t_k)\,\Delta t + \mathbf{L}\,\Delta\boldsymbol{\beta}_k \\ &\quad + \mathbf{a}_k(\hat{\mathbf{x}}(t_k))\,\frac{\Delta t^2}{2} + \sum_j \mathbf{b}_{k,j}(\hat{\mathbf{x}}(t_k))\,\Delta\boldsymbol{\zeta}_k,\end{aligned} \tag{9.34}$$

where the terms $\mathbf{a}_k(\hat{\mathbf{x}}(t_k))$ *and* $\mathbf{b}_{k,j}(\hat{\mathbf{x}}(t_k))$ *are given by Equation* (8.48) *and the terms* $\Delta\boldsymbol{\beta}_k, \Delta\boldsymbol{\zeta}_k$ *are jointly Gaussian as defined in Equation* (8.46). *If we put* $\mathbf{e}_k = (\Delta\boldsymbol{\beta}_k, \Delta\boldsymbol{\zeta}_k)$*, then the mean and covariance recursions can be written in form* (9.33)*, where the integrations can be approximated, for example, using cubature type of approximations discussed in the previous section.*

The preceding kind of approximation has been used in filtering context, for example, by Särkkä and Solin (2012), where the authors also compared it with the approximations described in the previous section.

9.3 Local Linearization Methods of Ozaki and Shoji

The local linearization methods of Ozaki (1992, 1993) and Shoji and Ozaki (1998) (see also Rao, 1999; Iacus, 2008) can be seen as instances of linearization methods, where the linearization is chosen in a specific way. The basic idea of the method of Ozaki (1993) can be described as follows. We aim to approximate the solution of the following scalar SDE:

$$\mathrm{d}x = f(x)\,\mathrm{d}t + \mathrm{d}\beta. \tag{9.35}$$

The idea is to approximate the solution on interval $[t, t + \Delta t]$, conditioned on $x(t)$ as a linear SDE

$$\mathrm{d}x = F\,x\,\mathrm{d}t + \mathrm{d}\beta. \tag{9.36}$$

The reason why this is useful is that the discretization for linear equation is then readily given by Equations (6.24) and (6.25) as

$$\begin{aligned} A(\Delta t) &= \exp(F\,\Delta t), \\ \Sigma(\Delta t) &= \int_0^{\Delta t} \exp(2F\,(\tau - \Delta t))\,q\,\mathrm{d}\tau \\ &= \frac{q}{2F}\,[\exp(2F\,\Delta t) - 1], \end{aligned} \tag{9.37}$$

giving the Gaussian approximation

$$p(x(t + \Delta t) \mid x(t)) \approx \mathrm{N}(x(t + \Delta t) \mid A(\Delta t)\,x(t), \Sigma(\Delta t)). \tag{9.38}$$

To derive the method, let us now consider the noise-free equation

$$\frac{\mathrm{d}\xi(t)}{\mathrm{d}t} = f(\xi), \quad \xi(t) = x(t). \tag{9.39}$$

Differentiating once gives

$$\frac{\mathrm{d}^2\xi}{\mathrm{d}t^2} = f'(\xi)\,\frac{\mathrm{d}\xi}{\mathrm{d}t}, \tag{9.40}$$

where f' is the derivative of f. Let us now assume that the derivative is constant with the value of $f'(x(t))$. Then, by solving the differential equation (9.40), we get

$$\frac{\mathrm{d}\xi}{\mathrm{d}t}(t + \Delta t) = \exp(f'(x(t))\,\Delta t)\,\frac{\mathrm{d}\xi(t)}{\mathrm{d}t} = \exp(f'(x(t))\,\Delta t)\,f(x(t)). \tag{9.41}$$

Integrating from 0 to Δt then gives

$$\begin{aligned}\xi(t + \Delta t) &= x(t) + \int_0^{\Delta t} \exp(f'(x(t))\,\tau)\,f(x(t))\,\mathrm{d}\tau \\ &= x(t) + \frac{1}{f'(x(t))}[\exp(f'(x(t))\,\Delta t) - 1]\,f(x(t)) \\ &= \left(1 + \frac{f(x(t))}{x(t)\,f'(x(t))}[\exp(f'(x(t))\,\Delta t) - 1]\right)\,x(t). \end{aligned} \tag{9.42}$$

We then finally wish to determine F such that $\exp(f'(x(t))\,\Delta t)$ matches the coefficient of $x(t)$ in the preceding, that is,

$$\exp(F\,\Delta t) = 1 + \frac{f(x(t))}{x(t)\,f'(x(t))}[\exp(f'(x(t))\,\Delta t) - 1], \tag{9.43}$$

which gives

$$F = \frac{1}{\Delta t}\log\left(1 + \frac{f(x(t))}{x(t)\,f'(x(t))}[\exp(f'(x(t))\,\Delta t) - 1]\right). \tag{9.44}$$

This method can also be extended to multivariate SDEs, but it becomes more complicated and less useful.

In the scalar case, we get the following algorithm.

Algorithm 9.11 (Local linearization of Ozaki)**.** *Given a scalar SDE of the form*

$$\mathrm{d}x = f(x)\,\mathrm{d}t + \mathrm{d}\beta, \tag{9.45}$$

we approximate its solution on interval $[t, t+\Delta t]$ *by a linear time-invariant SDE*

$$\mathrm{d}x = F\, x\, \mathrm{d}t + \mathrm{d}\beta, \tag{9.46}$$

where

$$F = \frac{1}{\Delta t} \log \left(1 + \frac{f(x(t))}{x(t)\, f'(x(t))} [\exp(f'(x(t))\, \Delta t) - 1] \right). \tag{9.47}$$

The method of Shoji and Ozaki (1998) is a modification to the local linearization method of Ozaki (1992, 1993), which extends more easily to multivariate case. We aim to approximate the solution of a scalar SDE

$$\mathrm{d}x = f(x,t)\, \mathrm{d}t + \mathrm{d}\beta \tag{9.48}$$

with a linear SDE

$$\mathrm{d}x = G\, x\, \mathrm{d}t + a\, t\, \mathrm{d}t + b\, \mathrm{d}t + \mathrm{d}\beta. \tag{9.49}$$

Recall that the Itô formula for f gives

$$\begin{aligned} \mathrm{d}f(x,t) &= \frac{\partial f(x,t)}{\partial t}\, \mathrm{d}t + \frac{\partial f(x,t)}{\partial x}\, \mathrm{d}x + \frac{1}{2}\, \frac{\partial^2 f(x,t)}{\partial x^2}\, \mathrm{d}x^2 \\ &= \left[\frac{\partial f(x,t)}{\partial t} + \frac{q}{2}\, \frac{\partial^2 f(x,t)}{\partial x^2} \right] \mathrm{d}t + \frac{\partial f(x,t)}{\mathrm{d}x}\, \mathrm{d}x. \end{aligned} \tag{9.50}$$

Let us now assume that $\partial f / \partial x$, $\partial^2 f / \partial x^2$, and $\partial f / \partial t$ are constant. Then on the interval $[t, u]$, this gives

$$\begin{aligned} f(x(u), u) - f(x(t), t) = \left[\frac{\partial f(x,t)}{\partial t} + \frac{q}{2}\, \frac{\partial^2 f(x,t)}{\partial x^2} \right] (u - t) \\ + \frac{\partial f(x,t)}{\partial x}\, (x(u) - x(t)). \end{aligned} \tag{9.51}$$

We get the approximation

$$f(x(u), u) = G(t)\, x(u) + a(t)\, u + b(t), \tag{9.52}$$

where the terms are given by

$$\begin{aligned} G(t) &= \frac{\partial f(x,t)}{\partial x}, \\ a(t) &= \frac{\partial f(x,t)}{\partial t} + \frac{q}{2}\, \frac{\partial^2 f(x,t)}{\partial x^2}, \\ b(t) &= f(x(t), t) - \frac{\partial f(x,t)}{\partial x}\, x(t) - \left[\frac{\partial f(x,t)}{\partial t} + \frac{q}{2}\, \frac{\partial^2 f(x,t)}{\partial x^2} \right] t. \end{aligned} \tag{9.53}$$

Here we can now pick up the coefficients for the approximation. In algorithmic form the method is the following.

Algorithm 9.12 (Local linearization of Shoji and Ozaki)**.** *Given a scalar SDE of the form*

$$\mathrm{d}x = f(x,t)\,\mathrm{d}t + \mathrm{d}\beta, \tag{9.54}$$

we approximate it on $[t, t + \Delta t]$ *is as a linear SDE*

$$\mathrm{d}x = G\,x\,\mathrm{d}t + a\,t\,\mathrm{d}t + b\,\mathrm{d}t + \mathrm{d}\beta, \tag{9.55}$$

where

$$\begin{aligned} G &= \frac{\partial f(x(t),t)}{\partial x}, \\ a &= \frac{\partial f(x(t),t)}{\partial t} + \frac{q}{2}\,\frac{\partial^2 f(x(t),t)}{\partial x^2}, \\ b &= f(x(t),t) - \frac{\partial f(x(t),t)}{\partial x}\,x(t) - \left[\frac{\partial f(x(t),t)}{\partial t} + \frac{q}{2}\,\frac{\partial^2 f(x(t),t)}{\partial x^2}\right] t. \end{aligned} \tag{9.56}$$

Due to the appearance of $x(t)$ in the linearization coefficients, local linearization methods are not suitable for forming Gaussian (process) approximations to SDEs in the same sense as Gaussian assumed density approximations are. However, they were originally developed for numerical simulation of SDEs as well as for one-step approximation of SDEs for parameter estimation. Numerical simulation with local linearization can be done analogously to Algorithm 9.9 – at each small subinterval of length Δt, we draw from the local linearization approximation instead of the true transition density. This can be written in the following algorithmic form.

Algorithm 9.13 (SDE simulation by local linearization)**.** *Local linearization methods can be used for simulating trajectories from an SDE as follows:*

1. *Divide the simulation interval* $[t_0, t]$ *into* M *subintervals of length* Δt.
2. *At each subinterval, sequentially draw a new sample from the linear SDE corresponding to the local linearization approximation.*

The following example provides a comparison of the local linearization methods with the Gaussian assumed density approximations outlined in Section 9.1.

Example 9.14 (Local linearization vs. Gaussian approximations)**.** *For example, consider the Beneš SDE*

$$\mathrm{d}x = \tanh(x)\,\mathrm{d}t + \mathrm{d}\beta, \quad x(0) = x_0. \tag{9.57}$$

The first local linearization in Algorithm 9.11 gives

$$\mathrm{d}x = F\,x\,\mathrm{d}t + \mathrm{d}\beta, \tag{9.58}$$

where

$$F = \frac{1}{\Delta t}\log\Bigg(1 + \frac{\tanh(x_0)}{x_0\,[1-\tanh^2(x_0)]} \\ \times [\exp([1-\tanh^2(x_0)]\,\Delta t) - 1]\Bigg). \tag{9.59}$$

The coefficients for the second linearization are

$$\begin{aligned} G &= 1 - \tanh^2(x_0), \\ a &= q\,\tanh(x_0)\,(\tanh^2(x_0) - 1), \\ b &= \tanh(x_0) - x_0\,(\tanh^2(x_0) - 1). \end{aligned} \tag{9.60}$$

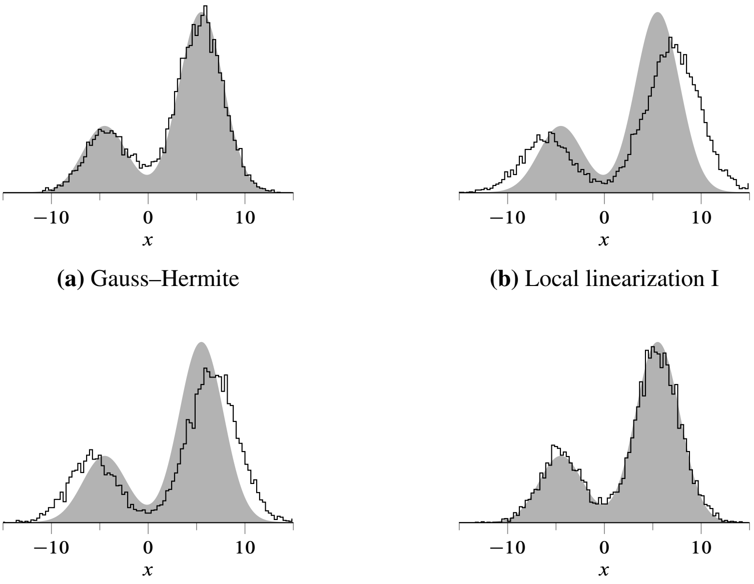
Also recall that the corresponding (classical) linearization and sigma-point approximations are given by Equations (9.23) *and* (9.24) *with initial conditions* $m(0) = x_0$ *and* $P(0) = 0$.

Figure 9.3 shows simulation results computed with the local linearization methods, the sigma point–based simulation method (cf. Example 9.6), and the strong Itô–Taylor method of order 1.5. *In order to test the differences between the methods, we computed the histograms of the simulation results at time* $t = 5$ *with initial condition* $x(0) = 1/2$ *and by using a relatively large* $\Delta t = 1$ *in each method. The sigma-point method was a Gauss–Hermite–based method with 10 points. As can be seen, in this case the sigma-point method produces a more accurate histogram than the local linearization methods.*

9.4 Taylor Series Expansions of Moment Equations

One way to approximate the moments of an SDEs on an interval $[t, t + \Delta t]$ is to use the Taylor series expansion of the function (Kessler, 1997; Aït-Sahalia, 2002, 2008)

$$\Delta t \mapsto \mathrm{E}[\phi(\mathbf{x}(t + \Delta t)) \mid \mathbf{x}(t)]. \tag{9.61}$$



(**a**) Gauss–Hermite

(**b**) Local linearization I

(**c**) Local linearization II

(**d**) Itô–Taylor

Figure 9.3 Comparison of numerical simulation of SDEs with local linearization methods in Example 9.14, the Gauss–Hermite–based simulation method, and the strong Itô–Taylor method of order 1.5. The solid density represents the ground-truth reference.

The Taylor series expansion centered at $\Delta t = 0$, is now given as

$$\mathrm{E}[\phi(\mathbf{x}(t + \Delta t))] = \sum_{k=0}^{\infty} \frac{1}{k!} \frac{\mathrm{d}^k \, \mathrm{E}[\phi(\mathbf{x}(t))]}{\mathrm{d}t^k} \, \Delta t^k, \tag{9.62}$$

for which we need the time derivatives of $\mathrm{E}[\phi(\mathbf{x}(t))]$. Fortunately, we know that the time derivative of this kind of expectation is given by the formula (5.13), which says that

$$\frac{\mathrm{d}\, \mathrm{E}[\phi(\mathbf{x}(t))]}{\mathrm{d}t} = \mathrm{E}[\mathcal{A}\phi(\mathbf{x}(t))], \tag{9.63}$$

where $\mathcal{A}$ is the generator of the diffusion. Applying Equation (9.63) again to the time derivative of $\mathcal{A}\,\phi$ then gives

$$\frac{\mathrm{d}\, \mathrm{E}[\mathcal{A}\phi(\mathbf{x}(t))]}{\mathrm{d}t} = \mathrm{E}[\mathcal{A}^2\phi(\mathbf{x}(t))], \tag{9.64}$$

and by using $\mathrm{E}[\mathcal{A}\phi(\mathbf{x}(t))] = \mathrm{d}\,\mathrm{E}[\phi(\mathbf{x}(t))]/\mathrm{d}t$ on the left-hand side, we get

$$\frac{\mathrm{d}^2\,\mathrm{E}[\phi(\mathbf{x}(t))]}{\mathrm{d}t^2} = \mathrm{E}[\mathcal{A}^2\phi(\mathbf{x}(t))]. \tag{9.65}$$

Applying this further to $\mathcal{A}^2\phi(\mathbf{x}(t)), \mathcal{A}^3\phi(\mathbf{x}(t)), \ldots$ then gives the general formula

$$\frac{\mathrm{d}^n\,\mathrm{E}[\phi(\mathbf{x}(t))]}{\mathrm{d}t^n} = \mathrm{E}[\mathcal{A}^n\phi(\mathbf{x}(t))]. \tag{9.66}$$

By using this, we can form the Taylor series expansion of $\mathrm{E}[\phi]$ as

$$\begin{aligned}\mathrm{E}[\phi(\mathbf{x}(t+\Delta t))] &= \sum_{k=0}^{\infty} \frac{1}{k!}\,\frac{\mathrm{d}^k\,\mathrm{E}[\phi(\mathbf{x}(t))]}{\mathrm{d}t^k}\,\Delta t^k \\ &= \sum_{k=0}^{\infty} \frac{1}{k!}\,\mathrm{E}[\mathcal{A}^k\phi(\mathbf{x}(t))]\,\Delta t^k.\end{aligned} \tag{9.67}$$

However, if we now condition on $\mathbf{x}(t)$, the expectations in the series disappear and we get the following algorithm.

Algorithm 9.15 (Taylor series expansion of moments)**.** *Given an SDE*

$$\mathrm{d}\mathbf{x} = \mathbf{f}(\mathbf{x},t)\,\mathrm{d}t + \mathbf{L}(\mathbf{x},t)\,\mathrm{d}\boldsymbol{\beta}, \tag{9.68}$$

the conditional expectation of a function $\phi(\mathbf{x})$ on interval $[t, t+\Delta t]$ can be approximated by truncating the Taylor series expansion

$$\mathrm{E}[\phi(\mathbf{x}(t+\Delta t)) \mid \mathbf{x}(t)] = \sum_{k=0}^{\infty} \frac{1}{k!}\,\mathcal{A}^k\phi(\mathbf{x}(t))\,\Delta t^k, \tag{9.69}$$

where $\mathcal{A}$ is the generator of the diffusion in Definition 5.2.

The Taylor series approximation can be used to form Gaussian approximations in the same way as we did in Section 9.2. The Taylor series provides asymptotically exact expressions for the mean and covariance, and hence they can be used to replace the discretization-based approximations that we used in Equation (9.33). These kinds of methods have been proposed, for example, in Kessler (1997), Rao (1999), and Iacus (2008).

The moment computation is illustrated in the following example, where the series expansion turns out to be finite and hence the moments can be computed exactly from the Taylor series.

Example 9.16 (Moments of the Beneš SDE)**.** *Let*

$$\mathrm{d}x = \tanh(x)\,\mathrm{d}t + \mathrm{d}\beta, \tag{9.70}$$

where $\beta(t)$ is a standard Brownian motion. In this case, we have

$$\mathcal{A} = \tanh(x)\,\frac{\partial}{\partial x} + \frac{1}{2}\,\frac{\partial^2}{\partial x^2}. \tag{9.71}$$

Applying this to function $\phi(x) = x$, then gives

$$\begin{aligned} \mathcal{A}^0\phi(x) &= x, \\ \mathcal{A}^1\phi(x) &= \tanh(x), \\ \mathcal{A}^n\phi(x) &= 0, \quad \textit{for } n \geq 2, \end{aligned} \tag{9.72}$$

and to function $\phi(x) = x^2$:

$$\begin{aligned} \mathcal{A}^0\phi(x) &= x^2, \\ \mathcal{A}^1\phi(x) &= 2\,x\,\tanh(x) + 1, \\ \mathcal{A}^2\phi(x) &= 2, \\ \mathcal{A}^n\phi(x) &= 0, \quad \textit{for } n \geq 3. \end{aligned} \tag{9.73}$$

For function $\phi(x) = x^3$, we get:

$$\begin{aligned} \mathcal{A}^0\phi(x) &= x^3, \\ \mathcal{A}^1\phi(x) &= 3\,x + 3\,x^2\,\tanh(x), \\ \mathcal{A}^2\phi(x) &= 6\,x + 6\,\tanh(x), \\ \mathcal{A}^3\phi(x) &= 6\,\tanh(x), \\ \mathcal{A}^n\phi(x) &= 0, \quad \textit{for } n \geq 4. \end{aligned} \tag{9.74}$$

Thus we get that Taylor series expansions for the first three noncentral moments of the SDE on interval $[t, t + \Delta t]$, conditioned on $x(t)$, are

$$\begin{aligned} \mathrm{E}[x(t + \Delta t) \mid x(t)] &= x(t) + \tanh(x(t))\,\Delta t, \\ \mathrm{E}[x^2(t + \Delta t) \mid x(t)] &= x^2(t) + 2\,x(t)\,\tanh(x(t))\,\Delta t + \Delta t + \Delta t^2, \\ \mathrm{E}[x^3(t + \Delta t) \mid x(t)] &= x^3(t) + 3\,x(t)\,\Delta t + 3\,x^2(t)\,\tanh(x(t))\,\Delta t \\ &\quad + 3\,x(t)\,\Delta t^2 + 3\,\tanh(x(t))\,\Delta t^2 + \tanh(x(t))\,\Delta t^3, \end{aligned} \tag{9.75}$$

which are indeed exact. If we used the first two moments in the preceding to form a Gaussian approximation as in Example 9.6, we would recover the exact Gaussian fit shown in Figure 9.1. However, this approximation could be improved by using higher-order moments to form a non-Gaussian density approximation.

Similarly to Gaussian assumed density approximations in Section 9.1 and local linearization methods in Section 9.3, the local transition density approximation implied by the Taylor series expansion can be used to construct SDE simulation methods (cf. Algorithms 9.9 and 9.13).

9.5 Hermite Expansions of Transition Densities

Provided that the transition density of the SDE is close to a normal distribution, we can use a Fourier–Hermite series to approximate it. This idea is presented by Aït-Sahalia (2002, 2008) and can also be found, for example, in Rao (1999) and Iacus (2008).

Consider the following SDE:

$$\mathrm{d}\mathbf{x} = \mathbf{f}(\mathbf{x})\,\mathrm{d}t + \mathbf{L}\,\mathrm{d}\boldsymbol{\beta}, \tag{9.76}$$

where $\boldsymbol{\beta}$ is a vector of Brownian motions with diffusion matrix $\mathbf{Q}$. We also need to assume that $\mathbf{L}\,\mathbf{Q}\,\mathbf{L}^\mathsf{T}$ is invertible. In the articles of Aït-Sahalia (2002, 2008), it was assumed that the equation had already been transformed into unit diffusion by the Lamperti transform, but here we will directly apply the method to the preceding SDE with nonunit diffusion.

We now aim to form an approximation to the transition density

$$p_{\mathbf{x}}(\mathbf{x}(t+\Delta t) \mid \mathbf{x}(t)), \tag{9.77}$$

where $\Delta t > 0$. Let us now define a "pseudonormalized" increment

$$\mathbf{z} = [\mathbf{L}\,\mathbf{Q}\,\mathbf{L}^\mathsf{T}\,\Delta t]^{-1/2}\,[\mathbf{x}(t+\Delta t) - \mathbf{x}(t)], \tag{9.78}$$

which should be quite close to a $\mathrm{N}(\mathbf{0},\mathbf{I})$ random variable. If we denote the probability density of $\mathbf{z}$ given $\mathbf{x}(t)$ as $q(\mathbf{z} \mid \mathbf{x}(t))$, then the transition density of the original process can be rewritten as

$$\begin{aligned} &p(\mathbf{x}(t+\Delta t) \mid \mathbf{x}(t)) \\ &\quad = |\mathbf{L}\,\mathbf{Q}\,\mathbf{L}^\mathsf{T}\,\Delta t|^{-1/2}\,q([\mathbf{L}\,\mathbf{Q}\,\mathbf{L}^\mathsf{T}\,\Delta t]^{-1/2}\,[\mathbf{x}(t+\Delta t) - \mathbf{x}(t)] \mid \mathbf{x}(t)). \end{aligned} \tag{9.79}$$

Let $\mathrm{H}_{\boldsymbol{\alpha}}(\mathbf{z})$ be the multivariate probabilists' Hermite polynomials with multi-indexing $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_n)$, which can be defined as follows:

$$\mathrm{H}_{\boldsymbol{\alpha}}(\mathbf{z}) = \mathrm{H}_{\alpha_1}(z_1) \times \mathrm{H}_{\alpha_2}(z_1) \times \cdots \times \mathrm{H}_{\alpha_n}(z_n), \tag{9.80}$$

where H_j are univariate Hermite polynomials defined as

$$\mathrm{H}_j(z) = (-1)^j\,\exp(z^2/2)\,\frac{\mathrm{d}^j}{\mathrm{d}z^j}\exp(-z^2/2). \tag{9.81}$$

We can now form the Jth order Fourier–Hermite series expansion of q/N as follows:

$$\frac{q(\mathbf{z} \mid \mathbf{x}(t))}{\mathrm{N}(\mathbf{z} \mid \mathbf{0}, \mathbf{I})} \approx \sum_{|\boldsymbol{\alpha}| \le J} \frac{c_{\boldsymbol{\alpha}}}{\boldsymbol{\alpha}!} \mathrm{H}_{\boldsymbol{\alpha}}(\mathbf{z}), \tag{9.82}$$

where

$$\begin{aligned} c_{\boldsymbol{\alpha}} &= \int \mathrm{H}_{\boldsymbol{\alpha}}(\mathbf{z})\, q(\mathbf{z} \mid \mathbf{x}(t))\, \mathrm{d}\mathbf{z} \\ &= \mathrm{E}\left[\mathrm{H}_{\boldsymbol{\alpha}}([\mathbf{L}\,\mathbf{Q}\,\mathbf{L}^{\mathsf{T}}\,\Delta t]^{-1/2}\,[\mathbf{x}(t+\Delta t) - \mathbf{x}(t)]) \mid \mathbf{x}(t)\right]. \end{aligned} \tag{9.83}$$

This series will converge provided that the tails of $q(\mathbf{z} \mid \mathbf{x}(t))$ are "thin" enough (see Aït-Sahalia, 2002, 2008, for mathematical details). We can then recover the transition density approximation from Equation (9.79) as

$$\begin{aligned} & p(\mathbf{x}(t+\Delta t) \mid \mathbf{x}(t)) \\ &= \frac{\mathrm{N}(\mathbf{z} \mid \mathbf{0}, \mathbf{I})}{|\mathbf{L}\,\mathbf{Q}\,\mathbf{L}^{\mathsf{T}}\,\Delta t|^{1/2}}\, q([\mathbf{L}\,\mathbf{Q}\,\mathbf{L}^{\mathsf{T}}\,\Delta t]^{-1/2}\,[\mathbf{x}(t+\Delta t) - \mathbf{x}(t)] \mid \mathbf{x}(t)) \\ &\approx \frac{\mathrm{N}(\mathbf{z} \mid \mathbf{0}, \mathbf{I})}{|\mathbf{L}\,\mathbf{Q}\,\mathbf{L}^{\mathsf{T}}\,\Delta t|^{1/2}} \sum_{|\boldsymbol{\alpha}| \le J} \frac{c_{\boldsymbol{\alpha}}}{\boldsymbol{\alpha}!} \mathrm{H}_{\boldsymbol{\alpha}}([\mathbf{L}\,\mathbf{Q}\,\mathbf{L}^{\mathsf{T}}\,\Delta t]^{-1/2}\,[\mathbf{x}(t+\Delta t) - \mathbf{x}(t)]), \end{aligned} \tag{9.84}$$

In order to implement the method, we also need a way to compute the expectations appearing in Equation (9.83). For that purpose Aït-Sahalia (2002, 2008) suggests to use the moment Taylor series expansion that was discussed in Section 9.4. By selecting the orders of the Taylor series and Hermite series suitably, we can control the approximation accuracy. The resulting expansion is "closed form" in the sense that it is possible to tabulate expansions such that we only need to plug in the derivatives of the drift. Such expansions with Taylor series of order three and Hermite polynomials of order six, for scalar SDEs, have been tabulated in Aït-Sahalia (2002).

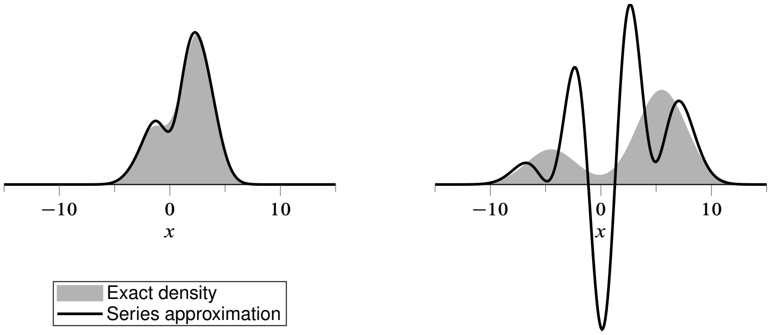
The algorithm is thus the following.

Algorithm 9.17 (Hermite expansion of transition density)**.** *The Hermite expansion approximation of the transition density of an SDE*

$$\mathrm{d}\mathbf{x} = \mathbf{f}(\mathbf{x})\,\mathrm{d}t + \mathbf{L}\,\mathrm{d}\boldsymbol{\beta} \tag{9.85}$$

can be formed as follows:

1. *Compute the coefficients $c_{\boldsymbol{\alpha}}$ in Equation* (9.83) *by using a truncated moment Taylor series expansion introduced in Section 9.4.*



(a) Hermite expansion at $t = 2$ **(b)** Hermite expansion at $t = 5$

Figure 9.4 Illustration of the Hermite expansion approximation in Example 9.18 for the Beneš SDE.

2. *Approximate the transition density of the SDE using a truncated Hermite expansion in Equation* (9.84).

The expansion is illustrated in the following example.

Example 9.18 (Hermite expansion of Beneš SDE)**.** *We used the coefficients tabulated in Aït-Sahalia (2002) to approximate the solution to the Beneš SDE*

$$\mathrm{d}x = \tanh(x)\,\mathrm{d}t + \mathrm{d}\beta, \quad x(0) = x_0. \tag{9.86}$$

Figure 9.4 shows the approximation at $t = 2$ and $t = 5$ with $x_0 = 1/2$. The approximation at $t = 2$ is still quite accurate, but the approximation at $t = 5$ is already quite inaccurate.

9.6 Discretization of FPK

We can also use partial differential equation (PDE) approximations to obtain approximate solutions to the Fokker–Planck–Kolmogorov PDE or its transition density. The Hermite-expansion and Gaussian approximations can indeed be seen as methods in this class, but here we consider the finite differences and basis function type of (Galerkin) approaches that can be seen as methods to discretize the FPK, that is, to approximate it as a finite-dimensional equation. More details on PDE methodology can be found, for example, in the books of Kreyszig (1993) and Brenner and Scott (2002).

Recall that the Fokker–Planck–Kolmorogov equation has the form

$$\frac{\partial p(\mathbf{x},t)}{\partial t} = \mathcal{A}^* p(\mathbf{x},t), \tag{9.87}$$

where $\mathcal{A}^*$ is the operator defined in Equation (5.36). It turns out that we can approximate this equation as a finite-dimensional system, which is just a homogeneous linear system that we can easily numerically solve. How to obtain this finite-dimensional system is, though, the key question, and here we take a look at two methods to do that: the finite-differences and basis function approximations.

We already saw in Exercise 5.3 that we can use finite differences to approximate the PDE. For that purpose, we need to discretize the state space to a finite grid $\{\mathbf{x}_i : i = 1,2,\ldots,N\}$ and then approximate the derivatives as finite differences. A similar finite-differences approximation was also used to generate the reference result in Figure 7.3. In the one-dimensional case, the approximation could be given as

$$\begin{aligned} \frac{\partial p(x,t)}{\partial x} &\approx \frac{p(x+h,t) - p(x-h,t)}{2h}, \\ \frac{\partial^2 p(x,t)}{\partial x^2} &\approx \frac{p(x+h,t) - 2p(x,t) + p(x-h,t)}{h^2}, \end{aligned} \tag{9.88}$$

and analogously in the multivariate case. If we collect the grid point evaluations $p(\mathbf{x}_i,t)$ into a vector $\mathbf{p}(t)$, this kind of approximation corresponds to certain matrix multiplication of $\mathbf{p}(t)$. Thus the whole FPK can be rewritten as a linear system

$$\frac{\mathrm{d}\mathbf{p}}{\mathrm{d}t} = \mathbf{F}_{\mathrm{fd}}(t)\,\mathbf{p}, \tag{9.89}$$

where $\mathbf{F}_{\mathrm{fd}}(t)$ is the finite-difference approximation matrix for the operator $\mathcal{A}^*$. An advantage of finite-difference approximations is that the matrix $\mathbf{F}_{\mathrm{fd}}(t)$ tends to be very sparse and hence sparse matrix routines can be used to make the computations faster.

Given the grid evaluations of the initial conditions $p(\mathbf{x}_i,t_0)$ collected into a vector $\mathbf{p}(t_0)$, this equation can now be solved as

$$\mathbf{p}(t) = \boldsymbol{\Psi}_{\mathrm{fd}}(t,t_0)\,\mathbf{p}(t_0), \tag{9.90}$$

where $\boldsymbol{\Psi}_{\mathrm{fd}}$ is the transition matrix corresponding to $\mathbf{F}_{\mathrm{fd}}(t)$. In a time-invariant case, we thus simply have

$$\mathbf{p}(t) = \exp((t-t_0)\,\mathbf{F}_{\mathrm{fd}})\,\mathbf{p}(t_0). \tag{9.91}$$

However, in practice we often use methods like backward Euler or Runge–Kutta for the temporal integration, because the computation of the transition matrix or the matrix exponential can be computationally heavy or even intractable. In particular, the matrix exponential tends to be a dense matrix even when $\mathbf{F}_{\mathrm{fd}}$ is a sparse matrix, and hence the speedup provided by sparse routines is lost. It is also possible to jointly discretize the time direction to get a single equation for the full space–time solution.

Another useful approach for the FPK solution is to fix a set of basis functions $\{\phi_i(\mathbf{x}) : i = 1, 2, \ldots\}$ and then approximate the PDE solution as

$$p(\mathbf{x}, t) \approx \sum_{i=1}^{N} p_i(t)\, \phi_i(\mathbf{x}), \tag{9.92}$$

where the time evolution of the coefficients $p_i(t)$ is to be determined – note that the coefficients are not usually the same as with the finite-difference approximation, although the finite-difference approximation is a special case of this kind of expansions.

It now turns out that we can approximate the operator $\mathcal{A}^*$ on the basis $\{\phi_i\}$ with a matrix $\mathbf{F}_{\mathrm{bf}}(t)$ that operates on the coefficients. The operation $\mathcal{A}^* p$ then corresponds to multiplication of the coefficients in the expansion (9.92) with this matrix:

$$\mathcal{A}^* p \leftrightarrow \mathbf{F}_{\mathrm{bf}}(t)\, \mathbf{p}, \tag{9.93}$$

where $\mathbf{p} = \big(p_1, p_2, \ldots\big)$ is a vector formed from the series expansion coefficients. Because the basis functions ϕ_i are independent of time, we also have the correspondence

$$\frac{\partial p}{\partial t} \leftrightarrow \frac{\mathrm{d}\mathbf{p}}{\mathrm{d}t}. \tag{9.94}$$

Thus we can express the FPK PDE on the basis $\{\phi_i\}$ as follows:

$$\frac{\mathrm{d}\mathbf{p}}{\mathrm{d}t} = \mathbf{F}_{\mathrm{bf}}(t)\, \mathbf{p}. \tag{9.95}$$

Note that this equation has the same form as Equation (9.89), and we can use either matrix exponential (in the time-independent case) or the transition matrix corresponding to $\mathbf{F}_{\mathrm{bf}}(t)$ to express the solution as

$$\mathbf{p}(t) = \boldsymbol{\Psi}_{\mathrm{bf}}(t, t_0)\, \mathbf{p}(t_0), \tag{9.96}$$

where $\mathbf{p}(t_0)$ is the vector of projection coefficients of $p(\mathbf{x}, t_0)$.

How can we obtain the matrix $\mathbf{F}_{\mathrm{bf}}(t)$ then? Let us put

$$g(\mathbf{x}) = \sum_{i=1}^{N} g_i \, \phi_i(\mathbf{x}) \quad \text{and} \quad f(\mathbf{x}) = \sum_{i=1}^{N} f_i \, \phi_i(\mathbf{x}), \tag{9.97}$$

and consider the transformation $g = \mathcal{A}^* f$, which now looks like this:

$$\sum_{i=1}^{N} g_i \, \phi_i(\mathbf{x}) = \sum_{i=1}^{N} f_i \, \mathcal{A}^* \phi_i(\mathbf{x}). \tag{9.98}$$

Assume that we know $\{f_i\}$ and wish to determine $\{g_i\}$, and denote the corresponding vectors as $\mathbf{f}$ and $\mathbf{g}$. We now have a few options to determine the required matrix:

Point collocation: One way is to select N points $\{\mathbf{x}_j\}$ in the state space and insist that the equation must hold on each of them:

$$\begin{gathered}\sum_{i=1}^{N} g_i \, \phi_i(\mathbf{x}_1) = \sum_{i=1}^{N} f_i \, \mathcal{A}^* \phi_i(\mathbf{x}_1), \\ \vdots \\ \sum_{i=1}^{N} g_i \, \phi_i(\mathbf{x}_N) = \sum_{i=1}^{N} f_i \, \mathcal{A}^* \phi_i(\mathbf{x}_N).\end{gathered} \tag{9.99}$$

If we define matrices $\mathbf{\Phi}$ and $\mathbf{\Gamma}$ such that $\Phi_{ij} = \phi_j(\mathbf{x}_i)$, $\Gamma_{ij} = \mathcal{A}^* \phi_j(\mathbf{x}_i)$, then this equation has the form

$$\mathbf{\Phi} \, \mathbf{g} = \mathbf{\Gamma} \, \mathbf{f}, \quad \text{which gives} \quad \mathbf{g} = \mathbf{\Phi}^{-1} \, \mathbf{\Gamma} \, \mathbf{f} \tag{9.100}$$

and thus the matrix is determined via $\mathbf{F}_{\mathrm{bf}}(t) = \mathbf{\Phi}^{-1} \, \mathbf{\Gamma}$.

Ritz–Galerkin method: We can also fix an inner product $\langle \bullet, \bullet \rangle$ and choose another set of N functions $\{\varphi_i\}$. We can now take inner products of Equations (9.98) with each of these functions to give

$$\begin{gathered}\sum_{i=1}^{N} g_i \, \langle \phi_i, \varphi_1 \rangle = \sum_{i=1}^{N} f_i \, \langle \mathcal{A}^* \phi_i, \varphi_1 \rangle, \\ \vdots \\ \sum_{i=1}^{N} g_i \langle \phi_i, \varphi_N \rangle = \sum_{i=1}^{N} f_i \, \langle \mathcal{A}^* \phi_i, \varphi_N \rangle.\end{gathered} \tag{9.101}$$

Defining matrices $\mathbf{\Phi}$ and $\mathbf{\Gamma}$ via $\Phi_{ij} = \langle \phi_j, \varphi_i \rangle$ and $\Gamma_{ij} = \langle \mathcal{A}^* \phi_j, \varphi_i \rangle$, this again leads to the matrix

$$\mathbf{F}_{\mathrm{bf}}(t) = \mathbf{\Phi}^{-1} \, \mathbf{\Gamma}. \tag{9.102}$$

Other basis function methods: We can also get several other methods as special cases of the Ritz–Galerkin method:

1. A very common choice of basis functions is to put $\varphi_i = \phi_i$ and select the basis functions to be orthonormal $\langle \phi_i, \phi_j \rangle = \delta_{ij}$. This class of methods is often called spectral methods. An example of such basis is the canonical Fourier basis consisting of sines and cosines. Due to the orthonormality, the matrix $\mathbf{F}_{\mathrm{bf}}(t)$ directly consists of the elements $\langle \mathcal{A}^* \phi_j, \phi_i \rangle$.
2. Using basis functions or "elements" that have a finite support leads to finite-element methods (FEM), which are very popular in engineering applications. However, in order to allow for simpler (typically piecewise linear) basis functions, we need to integrate $\langle \mathcal{A}^* \phi_j, \varphi_i \rangle$ by parts to have only single differentiation on both ϕ_j and φ_i, which complicates the equations a bit. For details, see Brenner and Scott (2002). The advantage of FEM is that it is suitable for sparse matrix routines, as the finite support of the elements induces sparseness in the matrices.
3. The point collocation corresponds to the selection $\varphi_i(\mathbf{x}) = \delta(\mathbf{x} - \mathbf{x}_i)$ with an L_2 inner product.

We can now formulate the following generic algorithm.

Algorithm 9.19 (Discretization approximation of FPK). *The following is a generic algorithm for discretization approximation of FPK targeted for solving $p(\mathbf{x}, t)$ given $p(\mathbf{x}, t_0)$ where $t > t_0$:*

1. *Construct a finite-dimensional vector $\mathbf{p}(t_0) = \big(p_1(t_0), \ldots, p_N(t_0)\big)$ such that we can represent the initial $p(\mathbf{x}, t_0)$ approximately as a series expansion*

$$p(\mathbf{x}, t_0) \approx \sum_{i=1}^{N} p_i(t_0) \, \phi_i(\mathbf{x}). \tag{9.103}$$

The basis functions can be piecewise constant (as in finite differences), Fourier basis functions, or any of the other Galerkin or finite-element type of basis functions discussed previously.

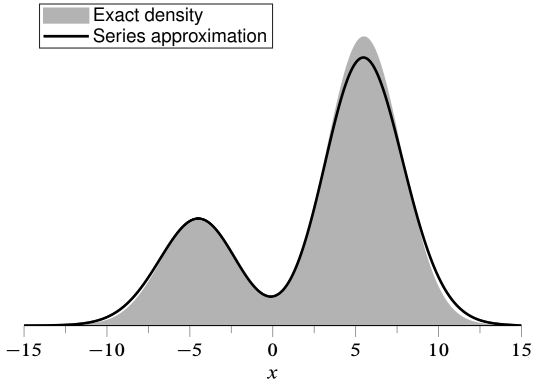


Figure 9.5 Illustration of the series expansion of FPK equation for the Beneš SDE in Example 9.20.

2. *Construct a discretization matrix* $\mathbf{F}(t)$ *by projecting or otherwise approximating the operator on the basis using the previously described methods.*
3. *Solve the linear system*

$$\frac{\mathrm{d}\mathbf{p}}{\mathrm{d}t} = \mathbf{F}(t)\,\mathbf{p}, \quad \mathbf{p}(t_0) = \textit{given}, \tag{9.104}$$

at time t *using backward Euler or Runge–Kutta or by computing the matrix exponential in the case of a time-invariant system (and if the nonsparseness is not an issue).*

4. *Approximate the probability density at* t *as*

$$p(\mathbf{x}, t) \approx \sum_{i=1}^{N} p_i(t)\,\phi_i(\mathbf{x}). \tag{9.105}$$

If we are interested in the transition density, then the initial coefficients $\mathbf{p}(t_0)$ *should be selected to approximate the Dirac delta function.*

An example of FPK approximation using an orthonormal basis is presented in the following.

Example 9.20 (Discretized FPK for the Beneš SDE). *Consider the model*

$$\mathrm{d}x = \tanh(x)\,\mathrm{d}t + \mathrm{d}\beta, \tag{9.106}$$

which gives the FPK

$$\frac{\partial p(x,t)}{\partial t} = -\frac{\partial}{\partial x}\left[\tanh(x)\, p(x,t)\right] + \frac{1}{2}\frac{\partial^2 p(x,t)}{\partial x^2}. \tag{9.107}$$

Let us select the basis

$$\phi_i(x) = \frac{1}{\sqrt{L}} \sin(\lambda_i\ (x+L)), \tag{9.108}$$

where $\lambda_i = \frac{i\,\pi}{2L}$, which is also the eigenbasis of the operator $\partial^2 \big/ \partial x^2$ with Dirichlet boundary conditions at $x = -L$ and $x = L$. If we define

$$\langle f, g \rangle = \int_{-L}^{L} f(x)\, g(x)\, \mathrm{d}x, \tag{9.109}$$

then $\langle \phi_i, \phi_j \rangle = \delta_{ij}$ and the matrix elements are given by

$$\begin{aligned}
[F_{\mathrm{bf}}]_{ij} &= \int_{-L}^{L} \left(-\frac{\partial}{\partial x}\left[\tanh(x)\, \phi_j(x)\right] + \frac{1}{2}\frac{\partial^2 \phi_j(x)}{\partial x^2} \right) \phi_i(x)\, \mathrm{d}x \\
&= \int_{-L}^{L} \left([\tanh^2(x) - 1]\, \phi_j(x) - \tanh(x)\, \frac{\partial \phi_j(x)}{\partial x} \right) \phi_i(x)\, \mathrm{d}x - \frac{1}{2}\lambda_i^2\, \delta_{ij} \\
&= \frac{1}{L} \int_{-L}^{L} \Big([\tanh^2(x) - 1]\ \sin(\lambda_j\ (x+L)) \\
&\qquad\qquad - \tanh(x)\, \lambda_j\ \cos(\lambda_j\ (x+L)) \Big) \sin(\lambda_i\ (x+L))\, \mathrm{d}x \\
&\quad - \frac{1}{2}\lambda_i^2\, \delta_{ij},
\end{aligned} \tag{9.110}$$

where we have used $\langle\, \partial^2 \phi_j \big/ \partial x^2\ , \phi_i \rangle = -\lambda_i^2 \delta_{ij}$ and the remaining integral can be evaluated with numerical integration.

The series expansion approximation with 20 terms is shown in Figure 9.5. The linear system resulting from the discretization was solved using a matrix exponential. As can be seen in the figure, the approximation is already quite close to the exact density. However, there is one disadvantage: the approximation becomes negative at some points and hence the interpretation of it directly as a probability density approximation can be hard.

9.7 Simulated Likelihood Methods

The basic idea in simulated likelihood methods (Pedersen, 1995; Brandt and Santa-Clara, 2002; Iacus, 2008) is to use a simple simulation to approximate the transition density $p(\mathbf{x}(t) \mid \mathbf{x}(t_0))$. We first divide the interval $[t_0, t]$ into M steps of length $(t - t_0)/M$. Then we use an Euler–Maruyama approximation on each subinterval. Instead of running Euler–Maruyama over all the intervals, we stop at step $m = M - 1$. We then compute the expectation of the Euler–Maruyama induced transition density approximation over that last subinterval. That is, we replace the last step in the Euler–Maruyama approximation with the transition density implied by the Euler approximation. This simple trick allows us to obtain a continuous approximation to the transition density which can be shown to converge to the true density (Pedersen, 1995).

The algorithm is the following.

Algorithm 9.21 (Simulated likelihood method)**.** *The transition density approximation in the simulated likelihood method is formed as follows:*

1. *Divide the interval $[t_0, t]$ to M subintervals of length Δt.*
2. *Run Euler–Maruyama over $M - 1$ steps, that is, until the second-to-last step $t_{M-1} = t - \Delta t$.*
3. *Perform the preceding simulation N times, which results in the samples $\hat{\mathbf{x}}^{(n)}(t_{M-1})$ for the second-to-last step, where $n = 1, \ldots, N$.*
4. *The transition density is then approximated as*

$$\begin{aligned} &p(\mathbf{x}(t) \mid \mathbf{x}(t_0)) \\ &\quad \approx \frac{1}{N} \sum_{n=1}^{N} \mathrm{N}(\mathbf{x}(t) \mid \hat{\mathbf{x}}^{(n)}(t_{M-1}) + \mathbf{f}(\hat{\mathbf{x}}^{(n)}(t_{M-1}), t_{M-1})\, \Delta t, \\ &\qquad \mathbf{L}(\hat{\mathbf{x}}^{(n)}(t_{M-1}), t_{M-1})\, \mathbf{Q}\, \mathbf{L}^{\mathsf{T}}(\hat{\mathbf{x}}^{(n)}(t_{M-1}), t_{M-1})\, \Delta t). \end{aligned} \tag{9.111}$$

Instead of the applying Euler–Maruyama approximation, we can also use any of the other SDE discretization methods from Chapter 8 that allow for closed-form transition density approximation. The approach can also be seen as a kernel density estimator, and we can replace the transition density–based kernel with some other kernel while still getting convergence of the approximation.

9.8 Pathwise Series Expansions and the Wong–Zakai Theorem

As discussed in Section 7.2, if we fix the time interval $[0, T]$, then on that interval standard Brownian motion has a series expansion of the form (see, e.g., Luo, 2006, and Section 7.2)

$$\beta(t) = \sum_{n=1}^{\infty} z_n \int_0^t \varphi_n(\tau) \, \mathrm{d}\tau, \tag{9.112}$$

where $z_n \sim \mathrm{N}(0, 1)$ for $n = 1, 2, \ldots$ are independent Gaussian random variables and $\{\varphi_n(t)\}$ are a suitable set of basis functions. One possible choice is the set of functions corresponding to the Karhunen–Loeve expansion given in Equation (7.29). The series expansion can be interpreted as the following representation for the differential of standard Brownian motion:

$$\mathrm{d}\beta(t) = \sum_{n=1}^{\infty} z_n \, \varphi_n(t) \, \mathrm{d}t. \tag{9.113}$$

We can now consider approximating the following equation by substituting a finite number N of terms from the preceding sum for the term $\mathrm{d}\beta(t)$ in the scalar SDE

$$\mathrm{d}x = f(x, t) \, \mathrm{d}t + L(x, t) \, \mathrm{d}\beta. \tag{9.114}$$

In the limit $N \to \infty$, we could then expect to get the exact solution. However, it has been shown by Wong and Zakai (1965) that this approximation actually converges to the *Stratonovich* SDE

$$\mathrm{d}x = f(x, t) \, \mathrm{d}t + L(x, t) \circ \mathrm{d}\beta. \tag{9.115}$$

That is, we can approximate the preceding Stratonovich SDE with an equation of the form

$$\mathrm{d}x = f(x, t) \, \mathrm{d}t + L(x, t) \sum_{n=1}^{N} z_n \, \varphi_n(t) \, \mathrm{d}t, \tag{9.116}$$

which actually is just an ordinary differential equation

$$\frac{\mathrm{d}x}{\mathrm{d}t} = f(x, t) + L(x, t) \sum_{n=1}^{N} z_n \, \varphi_n(t), \tag{9.117}$$

and the solution converges to the exact solution, when $N \to \infty$. The solution of an Itô SDE can be approximated by first converting it into the corresponding Stratonovich equation and then approximating the resulting equation.

Now an obvious extension is to consider a multivariate version of this approximation. Because any multivariate Brownian motion can be formed as a linear combination of independent standard Brownian motions, it is possible to form analogous multivariate approximations. Unfortunately, in the multivariate case the approximation does not generally converge to the Stratonovich solution. There exists basis functions for which this is true (e.g., Haar wavelets), but the convergence is not generally guaranteed. Anyway, we get the following algorithm (cf. Lyons et al., 2012, 2014).

Algorithm 9.22 (Pathwise series expansion of SDEs). *An approximation to a Stratonovich SDE*

$$\mathrm{d}\mathbf{x} = \mathbf{f}(\mathbf{x}, t)\,\mathrm{d}t + \mathbf{L}(\mathbf{x}, t) \circ \mathrm{d}\boldsymbol{\beta}, \tag{9.118}$$

via a series expansion of the Brownian motion can be formed as follows:

1. *Approximate the vector of Brownian motions with a series expansion of the form*

$$\boldsymbol{\beta}(t) = \sqrt{\mathbf{Q}} \sum_{n=1}^{N} \mathbf{z}_n \int_0^t \varphi_n(\tau)\,\mathrm{d}\tau, \tag{9.119}$$

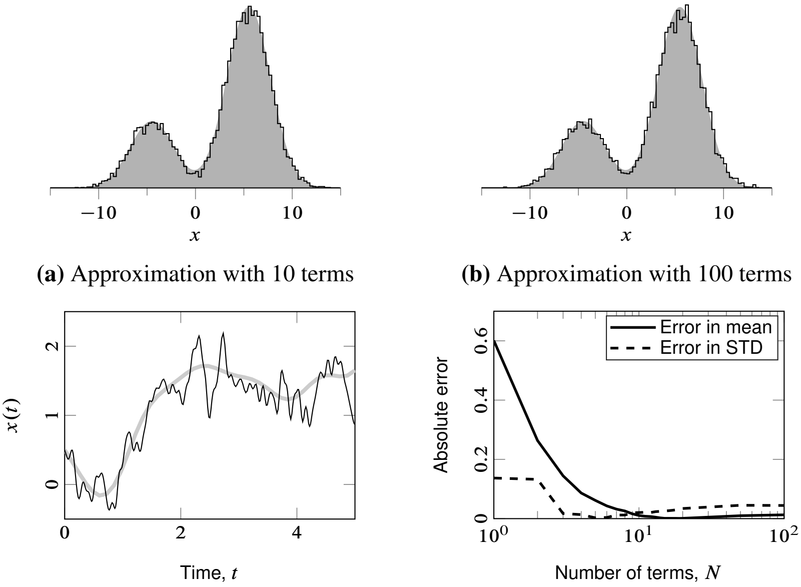
   where $\mathbf{z}_n \sim \mathrm{N}(\mathbf{0}, \mathbf{I})$ *are independent Gaussian random variables. For example, we can select* $\varphi_n(t)$ *to be the Fourier cosine basis in* (7.29).
2. *The corresponding approximation to the SDE is then given by the ordinary differential equation*

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \mathbf{f}(\mathbf{x}, t) + \mathbf{L}(\mathbf{x}, t)\,\sqrt{\mathbf{Q}} \sum_{n=1}^{N} \mathbf{z}_n\,\varphi_n(t). \tag{9.120}$$

However, the convergence of the preceding algorithm with $N \to \infty$ needs to be carefully investigated case-by-case (Lyons et al., 2014). Also recall that in the additive noise case the Itô and Stratonovich SDEs are equivalent, and hence in that case, if the approximation converges, it converges to the corresponding Itô SDE.

The series expansion approximation can be used both for forming parametric approximations to SDE solutions as well as for simulating trajectories from SDEs.

Another type of series expansion is the so-called *Wiener chaos expansion* (see, e.g., Cameron and Martin, 1947; Luo, 2006). Assume that we indeed are able to solve the Equation (9.120) with any given countably infinite number of values $\{z_1, z_2, \ldots\}$. Then we can see the solution as a



(a) Approximation with 10 terms

(b) Approximation with 100 terms

(c) Trajectories with 10 and 100 terms

(d) Approximation error

Figure 9.6 The subfigures (a) and (b) show the probability density approximations resulting from the series expansions in the Beneš SDE in Example 9.23 with $N = 10$ and $N = 100$ terms. The subfigure (c) shows the corresponding trajectory approximations and (d) illustrates the accuracy of the approximation by showing the errors in the empirical approximations of the mean and standard deviation (STD).

function (or functional) of the form

$$x(t) = U(t; z_1, z_2, \dots). \tag{9.121}$$

The Wiener chaos expansion is the multivariate Fourier–Hermite series for the right-hand side. That is, it is a polynomial expansion of a generic functional of Brownian motion in terms of Gaussian random variables. Hence the expansion is also called *polynomial chaos.*

Example 9.23 (Pathwise series expansion of Beneš SDE). *The series expansion approximation of the Beneš SDE*

$$\mathrm{d}x = \tanh(x)\,\mathrm{d}t + \mathrm{d}\beta, \quad x(0) = x_0, \tag{9.122}$$

with $x_0 = 1/2$ at time $t = 5$, is shown in Figure 9.6. The series expansion *was the cosine series given in Equation* (7.29) *with* $T = 5$*, which thus corresponds to the approximation*

$$\frac{\mathrm{d}x}{\mathrm{d}t} = \tanh(x) + \sum_{n=1}^{N} z_n \left(\frac{2}{T}\right)^{1/2} \cos\left(\frac{(2n-1)\,\pi}{2T}\,t\right), \tag{9.123}$$

where $z_n \sim \mathrm{N}(0, 1)$. *As can be seen from the figure, the distribution of the SDE is already well approximated with* $N = 10$ *basis functions, although the trajectories themselves are much smoother than actual Brownian paths.*

9.9 Exercises

9.1 Gaussian approximation of SDEs:

(a) Form a Gaussian assumed density approximation to the SDE in Equation (8.134) in the time interval $t \in [0, 5]$ and compare it to the exact solution. Compute the Gaussian integrals numerically on a uniform grid.

(b) Form a Gaussian assumed density approximation to Equation (8.133) and numerically compare it to the histogram obtained in Exercise 8.1.

9.2 Derive a similar series of moments as in Example 9.16 for the sine diffusion model

$$\mathrm{d}x = \sin(x)\,\mathrm{d}t + \mathrm{d}\beta, \tag{9.124}$$

where $\beta(t)$ is a standard Brownian motion, and compare it to moments computed from the Euler–Maruyama method.

9.3 Derive the local linearization for the model given in Equation (9.124) and numerically compare it to a cubature-based sigma-point approximation.

9.4 Use a finite-differences approximation to the FPK equation for the model given in Equation (9.124):

(a) Select the same basis as in Example 9.20 and write out the evolution equation.

(b) Solve it numerically and sketch the evolution of the probability density with a suitably chosen initial condition. Compare the result to the Euler–Maruyama method.

9.5 Implement a pathwise series approximation to the model (9.124) using the same basis as in Example 9.23.