THE UNIVERSITY of EDINBURGH

Inference and Parameter Estimation for Diffusion Processes

Simon Lyons

Doctor of Philosophy

Institute for Adaptive and Neural Computation

School of Informatics

University of Edinburgh

2014

Abstract

Diffusion processes provide a natural way of modelling a variety of physical and economic phenomena. It is often the case that one is unable to observe a diffusion process directly, and must instead rely on noisy observations that are discretely spaced in time. Given these discrete, noisy observations, one is faced with the task of inferring properties of the underlying diffusion process. For example, one might be interested in inferring the current state of the process given observations up to the present time (this is known as the filtering problem). Alternatively, one might wish to infer parameters governing the time evolution the diffusion process.

In general, one cannot apply Bayes' theorem directly, since the transition density of a general nonlinear diffusion is not computationally tractable. In this thesis, we investigate a novel method of simplifying the problem. The stochastic differential equation that describes the diffusion process is replaced with a simpler ordinary differential equation, which has a random driving noise that approximates Brownian motion. We show how one can exploit this approximation to improve on standard methods for inferring properties of nonlinear diffusion processes.

Acknowledgements

Firstly, I would like to thank my family for their continued support throughout my studies. Many thanks to the staff of iANC for their tireless efforts to help their students. Thanks also to my fellow PhD students for their consistently cheerful demeanour, and for teaching me to make really excellent coffee. Thanks to Microsoft Research for their financial support, and finally, thanks to Amos and Simo, without whom none of this would have been possible.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Simon Lyons*)

Table of Contents

Part I

Preliminaries

Chapter 1

Introduction

1.1 Motivation

Mathematical models based on ordinary differential equations (ODEs) have had an enormous effect on the development of modern science. Such models provide a means of quantifying the behaviour of physical systems, and thus making predictions about them. However, not all systems can be described adequately by ODEs. Small-scale systems are often subject to random effects that are difficult to model deterministically. This failure to capture noisy behaviour motivated the development of a stochastic counterpart to the ODE – the *stochastic differential equation* (SDE). An ODE gives a recipe for constructing a function by specifying how the function evolves over time. A stochastic differential equation does the same thing, but adds some form of noise to the evolution. Informally, we can write

$$\frac{d\mathbf{X}_t}{dt} = \mathbf{a}(\mathbf{X}_t) + \text{‘noise’}. \tag{1.1}$$

.

Figure 1.1 shows recorded Canadian lynx and snowshoe hare populations from 1845-1935 [1], alongside an ODE model and an SDE model of the predator-prey system. The ODE model captures the oscillatory behaviour of the population cycles, but the output is too regular to seem plausible. By adding some noise to the ODE model, we observe behaviour that is closer to the behaviour of the original system.

One sensible way to model the noise in (1.1) is to use *Gaussian white noise*. SDE models that use Gaussian white noise are flexible enough that they can capture a wide range of phenomena while still retaining a number of properties that make them tractable. Heuristically, Gaussian white noise induces random, normally-distributed

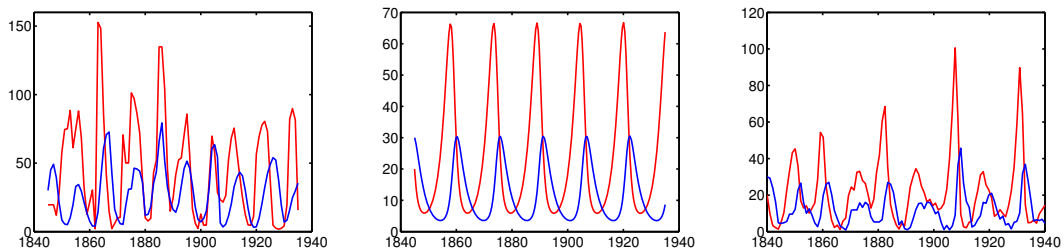


Figure 1.1: Predator-prey system showing populations of Canadian lynx (blue) and snowshoe hare (red). (left) Population data recorded from 1835 to 1937. (center) ODE model of the system. (right) SDE model of the system. We argue that the SDE model looks more 'natural' than the ODE model.

perturbations into the dynamics of a system. Over a short time Δt, these perturbations are comparable in size to $\sqrt{\Delta t}$. In other words, they are large compared to the timescale on which the system changes. However, the perturbations are also *independent*. This means that the perturbation over a short time interval $t_k - t_{k-1}$ will often cancel with the perturbation over $t_{k+1} - t_k$. The net result is that the perturbations cancel in just such a way that the solution to the SDE is a continuous, but nowhere-differentiable random function. At first glance, this seems almost miraculous, but we will see that there are good reasons explaining why it should be the case. SDEs that are driven by Gaussian white noise are usually referred to as *diffusion processes*.

Diffusion processes have been used to model prices of financial instruments [2], chemical reactions [3], firing patterns of individual neurons [4], weather patterns [5] and fMRI data [6], [7] among many other phenomena.

The analysis of diffusions dates back to Feller and Kolmogorov, who studied them as the scaling limits of certain Markov processes (see [8]). The theory of diffusion processes was revolutionised by Ito, who interpreted a diffusion process as the solution to a stochastic differential equation [9] [10].

1.2 Bayesian inference

It is often the case that one cannot observe a diffusion process $\mathbf{X}$ directly. For example, one might model the trajectory of an aircraft that is being buffeted by the wind as a diffusion. Observations of the aircraft's trajectory might come from a radar dish that rotates periodically. In such a setting, observations of the position of the aircraft

would arrive discretely in time, and may be subject to measurement error from the radar dish. Note that such a measurement system would record the position, but not the velocity of the aircraft. It is natural to report radar measurements in spherical polar coordinates. On the other hand, it is arguably more natural to develop a model of the aircraft dynamics based on rectangular Cartesian coordinates. For this reason, one should be prepared to model observations of the aircraft as *discrete-time, noisy* and possibly arising as a *nonlinear function of some or all of the state variables*.

The aim of this thesis is to explore how one can infer properties of a system based on incomplete, noisy, discrete-time observations in a way that is computationally efficient. In this thesis, we adopt the Bayesian formulation of statistical inference. We argue that the Bayesian framework is appropriate in the context of mathematical modelling using SDEs. One can 'build in' domain-specific knowledge while incorporating uncertainty over parameters that govern the dynamics of the system.

Our ultimate goal is to infer the posterior distribution of some statistic ζ of the system, based on prior knowledge of ζ and a sequence of observations $\{\mathbf{Y}_{t_k}\}_{1\leq k\leq n}$. We might aim to infer the position of the signal at some time t, so that $\zeta = \mathbf{X}_t$. Alternatively, we might aim to learn about some parameters $\zeta = \theta$ that govern the dynamics of the system: for example, one might be able to write down a diffusion model of a chemical reaction [3] without knowing exactly what the reaction rates are. In any event, we wish to compute the posterior distribution $p(\zeta \mid \mathbf{Y}_{t_1}, \ldots \mathbf{Y}_{t_n})$.

When $\zeta = \mathbf{X}_t$, it is sometimes the case that we have access to the observations *up to time t only*. That is, $t_n \leq t$. In this scenario, one is typically working in real time, with the aim of estimating $p(\mathbf{X}_t|\mathbf{Y}_{t_1}, \ldots, \mathbf{Y}_{t_n})$. However, as time progresses, new observations will arrive, and we need to *update* our posterior distribution of the state of $\mathbf{X}$. This is known as the *continuous-discrete filtering problem*. On the other hand, one might have access to an entire sequence of observations, so that $t_1 \leq t \leq t_n$. The problem of post-hoc estimation of the state of $\mathbf{X}_t$ given a batch of observations is known as the *continuous-discrete smoothing problem*.

Chapter 2

Brownian motion and SDEs

We will begin this Chapter by introducing Brownian motion - a stochastic process of fundamental importance in probability theory. We describe how Brownian motion is used a s a 'driving noise' in a stochastic differential equation, and give a short overview of stochastic integration and stochastic differential equations.

We then discuss the fundamental results in the theory of SDEs. In Section 2.2.3 we review Ito's lemma, which tells us how a SDE behaves under smooth transformations. We then derive the Fokker-Planck equation, which describes the time evolution of the transition density of a SDE. A basic method of discretisation and simulation of SDEs is reviewed in Section 2.2.5. We show how the dynamics of a diffusion processes behave under certain types of conditioning in Section 2.2.6. Finally, in Section 2.2.7, we give an overview of Girsanov's theorem, which can be thought of as a form of importance sampling for diffusion processes.

2.1 Brownian motion

We begin our discussion by introducing a stochastic process known as Brownian motion, which is of fundamental importance in probability theory. Brownian motion is named after Robert Brown, who studied the tiny and apparently random motion of pollen grains suspended on the surface of water. The phenomenon had been observed by many others before Brown, but Brown's contribution was to provide concrete evidence that the motion was not biological in nature [11].

The first mathematical characterisation of Brownian motion is due to Einstein [12]. At the time, Einstein's aim was to find evidence for the existence of atoms: in theory, a small particle that collides with many individual atoms or molecules should undergo

Brownian motion. Einstein attempted to deduce the mass of water molecules from observable quantities such as the temperature and viscosity of the water. Einstein's work was theoretical, and it was only later that he discovered the experimental work of Robert Brown. Einstein used some ad-hoc justifications for his analysis: the first rigorous construction of Brownian motion is credited to Norbert Wiener. For this reason it is sometimes known as the *Wiener process*.

There are many equivalent ways of characterising Brownian motion. Perhaps the most accessible definition is that Brownian motion on a time interval $[0,T]$ is a *Gaussian process* with mean 0 and covariance function $k(s,t) = \mathrm{Min}(s,t)$ (here, s and t are times in the interval $[0,T]$).

For our purposes, a Gaussian process is a probability distribution over functions $f : [0,T] \to \mathbb{R}$, though this can be generalised to include other domains and ranges. In order to define a Gaussian process, we must specify a *mean function* $m : [0,T] \to \mathbb{R}$ and a covariance function $k : [0,T] \times [0,T] \to \mathbb{R}$. These are the analogues of the mean and covariance of a Gaussian distribution.

Suppose g is a draw from a Gaussian process with mean m and covariance k. Let $(t_1,\ldots,t_N)$ be any collection of times in $[0,T]$. We can evaluate the random function g at each of these times, creating a vector $G = (g(t_1),\ldots,g(t_N))$. The defining feature of a Gaussian process is that G is distributed according to a multivariate Gaussian distribution with mean $(m(t_1),\ldots,m(t_n))$ and covariance matrix $(k(t_i,t_j))_{1\le i,j\le N}$.

One consequence of Brownian motion having covariance function $\mathrm{Min}(s,t)$ is that its increments are *independent*. Let $s < t$ be two times in $[0,T]$. We will use the symbol W to refer to a sample path of Brownian motion, and W_t to refer to the value of W at time t. The covariance between W_s and the increment $W_t - W_s$ is

$$\mathbb{E}[W_s(W_t - W_s)] = \min(s,t) - \min(s,s) = 0. \tag{2.1}$$

Any collection of random variables that are jointly Gaussian and uncorrelated are also independent. Strictly speaking, we should show that *any* collection of increments is independent, but the analysis above demonstrates the general strategy for proving such a result.

From the definition, we can also see that the variance of a given increment satisfies

$$\mathbb{E}[(W_t - W_s)^2] = \min(t,t) + \min(s,s) - 2\min(s,t) = t - s. \tag{2.2}$$

Brownian motion is ubiquitous throughout nature. It can be seen in the motion of dust particles, the hunting patterns of sharks [13], and the movement of the super-

massive black hole at the center of the galaxy [14]. This ubiquity is a consequence of *Donsker's invariance principle* [15], a generalisation of the central limit theorem. Roughly speaking, one can construct a random walk, defined at times $\{\Delta t, 2\Delta t, \dots\}$ by

$$X_{n\Delta t} = X_{(n-1)\Delta t} + \xi_n, \qquad X_0 = 0, \tag{2.3}$$

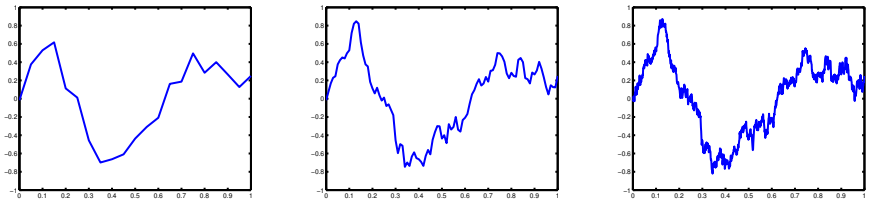


Figure 2.1: Random walk converging to Brownian motion

and at other times by linear interpolation. If the random variables ξ_i are i.i.d with finite variance, one can rescale the process so that it has variance 1 at time 1. In the limit as $\Delta t \to 0$, Donsker's invariance principle tells us that the rescaled random walk converges to Brownian motion. Thus, random walk-type behaviour that occurs on a suitable scale can often be approximated by Brownian motion.

It is worth emphasising that we are considering two subtly different processes here. Firstly, there is *physical* Brownian motion. That is, the random motion exhibited by, say, pollen grains suspended in water. Secondly, there is *mathematical* Brownian motion, which is an elegant model of the former process. It turns out that a particle with positive mass that is undergoing 'mathematical' Brownian motion would have infinite kinetic energy. This is related to the fact that, with probability 1, sample paths of Brownian motion are nowhere differentiable. Thus, the mathematical model is only applicable on a suitable timescale, and should not be taken literally.

2.1.1 L^2 spaces

We now aim to develop some basic tools that are useful for the study of Brownian motion. We will do so by analogy with concepts from linear algebra. One sensible way of learning linear algebra is to begin with the intuitive concepts of *lines*, *planes*, and *volumes* as model examples of vector spaces. More abstract vector spaces are then introduced, building on spatial intuition from the one-, two-, and three-dimensional cases.

The branch of mathematics known as *functional analysis* takes this abstraction one step further. The notion of 'point in a space' is now expanded to include functions, which are seen as 'points' in a 'function space'. One occasionally encounters the intuitive idea that 'functions are just infinitely long vectors'. Functional analysis provides a way of formalising this sentiment. We will limit our exposition to functions $f : [0,T] \to \mathbb{R}^N$ with $T > 0$, though it is possible to work far more generally than this.

We begin by observing that the familiar concepts from linear algebra can all be expressed in terms of functions. One can think of an N-dimensional vector v with real entries as a function $v : \{1,2,\ldots,N\} \to \mathbb{R}$. The first entry of v corresponds to $v(1)$, the second entry to $v(2)$, and so on.

Our aim is to build a 'propotypical' function space that shares as many properties as possible with Euclidean space. Arguably the simplest property that one can associate to a vector is some notion of length. A *norm* formalises the notion of length in a vector space. The standard Euclidean norm in N-dimensional space is given by

$$\|v\| = \left(\sum_{i=1}^{N} v(i)^2\right)^{1/2}. \tag{2.4}$$

If we try to replicate this definition with a function f, summing over all points in the domain $[0,T]$, we immediately see that the sum diverges. However, when we replace the sum by an integral, the result is 'well-behaved'. For this reason, we define a new norm by

$$\|f\| = \left(\int_0^T f^2(u)du\right)^{1/2}. \tag{2.5}$$

We refer to the set of all real-valued functions on $[0,T]$ that satisfy $\|f\| < \infty$ as $L^2([0,T];\mathbb{R})$, or $L^2(\mathbb{R})$ when there is no risk of ambiguity.

When $T = N$, and f is a piecewise constant function that jumps at $t = 1,2,\ldots,N-1$ we can regard f as an N-dimensional vector. In this case, the two norms give the same result regardless of whether we interpret f as a function or a vector. This suggests that the approach we have taken is, in some sense, the 'correct' one.

There are, however, some technicalities that one encounters when working with $L^2(\mathbb{R})$. In order to see this, we first define the *indicator function*

$$\begin{aligned} \mathbb{I}_{\{A\}}(x) &= 1 \qquad \text{if } x \in A \\ &= 0 \qquad \text{otherwise.} \end{aligned} \tag{2.6}$$

Intuitively, one would think of $f = \mathbb{I}_{\{[0,1]\}}(\cdot)$ and $g = \mathbb{I}_{\{(0,1)\}}(\cdot)$ (i.e. the indicator function over closed and open intervals respectively) as distinct functions. Here, we have

used the notation $f(\cdot)$ to emphasize that we are thinking of f as a function. Contrast this to the notation $f(x)$, which represents 'f evaluated at x' and is a real number). However, according to (2.5), $\|f-g\|=0$, so that f and g are in fact 'the same' function. The conclusion is that the L^2 norm cannot 'see' the behaviour of functions at individual points. The issue disappears if we are prepared only to consider the *average* behaviour of a function on a small interval $[t-\varepsilon, t+\varepsilon]$.

Building on the function/vector analogy, we consider the usual Euclidean inner product

$$\langle u, v\rangle = \sum_{i=1}^{N} u(i)v(i) \tag{2.7}$$

and generalise it to functions as before:

$$\langle f, g\rangle = \int_0^T f(u)g(u)du. \tag{2.8}$$

Vector spaces that have an inner product defined on them (and that also possess a technical property called *completeness*) are known as *Hilbert spaces*. The term is often reserved for infinite-dimensional vector spaces.

The inner product in $L^2(\mathbb{R})$ gives us a notion of *orthogonality of functions*. Recall that two vectors are defined to be orthogonal when their inner product is equal to 0. We use this same definition for functions. Any two functions that have *disjoint support* (that is, $\{x : f(x) \neq 0\}$ and $\{x : g(x) \neq 0\}$ have no elements in common) are orthogonal. A less obvious example is that $\sin(x)$ and $\sin(2x)$ are orthogonal on $[0, 2\pi]$.

In $\mathbb{R}^N$, we can find a collection of vectors $\{e_i\}_{i\leq N}$ such that $\|e_i\| = 1$, $\langle e_i, e_j\rangle = 0$ for $i \neq j$, and $\text{span}(\{e_i\}) = \mathbb{R}^N$. This is known as an *orthonormal basis*. Any vector v in $\mathbb{R}^N$ can be expressed in terms of a given orthonormal basis as follows:

$$v = \sum_{i=1}^{N} \langle v, e_i\rangle e_i. \tag{2.9}$$

As one might expect, one can also find orthonormal bases of $L^2([0,T];\mathbb{R})$. One important example is set of *Fourier sine* functions

$$\phi_1(t) = \frac{1}{\sqrt{T}} \tag{2.10}$$

$$\phi_k(t) = \sqrt{\frac{2}{T}} \sin\left(\frac{(k-1)\pi t}{T}\right), \qquad k > 1. \tag{2.11}$$

As before, when $f \in L^2(\mathbb{R})$, we can write it in terms of our basis as

$$f = \sum_{i=1}^{\infty} \langle f, \phi_i\rangle \phi_i. \tag{2.12}$$

We call this the *generalised Fourier series expansion* of f. We have added the term 'generalised' because we are using an arbitrary orthonormal basis. The standard Fourier series expansion uses a basis of Sine and Cosine functions.

The theory of *linear transforms* is central to linear algebra. Recall that a function $T : \mathbb{R}^N \to \mathbb{R}^M$ is a linear transform if, for all scalars α and β, and all vectors u and v,

$$T(\alpha u + \beta v) = \alpha T(u) + \beta T(v). \tag{2.13}$$

If we fix a basis, we can represent T using a matrix A with entries (a_{ij}). Just as we did with a vector, we can think of a matrix as a function $A : \{1,\ldots,N\} \times \{1,\ldots,M\} \to \mathbb{R}$. Setting $u = T(v)$, we have

$$u(i) = \sum_{j=1}^{N} A(i,j)v(j). \tag{2.14}$$

In an infinite-dimensional setting, one usually refers to a linear transform as a *linear operator*. The theory of linear operators on an infinite-dimensional space is considerably more complicated than in the finite-dimensional setting (Kreyszig [16] provides a friendly introduction to the subject. The classic reference is Dunford and Schwartz [17]).

In the majority of cases that are encountered in machine learning, linear operators behave as we would expect them to based on our intuition from $\mathbb{R}^N$. Just as we represented a linear operator in Euclidean space with a matrix, we represent a linear operator on a Hilbert space using a *kernel function*. We can think of a kernel as a function $k : [0,T] \times [0,T] \to \mathbb{R}$ (though the true definition is slightly more technical, including Dirac delta functions and related operators).

A kernel acts on a function f in much the same way as a matrix acts on a vector v. If $g = T(f)$, then

$$g(s) = \int_0^T k(s,t)f(t)dt. \tag{2.15}$$

If we extend our definition of 'kernel function' to include generalised functions such as the Dirac delta function, then the *Schwarz kernel theorem* [17] says that all linear operators admit a representation as a kernel function (it is counterintuitive, but even the derivative operator can be written as an integral operator).

The *eigenvectors* of a matrix A are those non-zero vectors that satisfy

$$\sum_{j=1}^{N} A(i,j)v(j) = \lambda v(i). \tag{2.16}$$

for some scalar λ. Similarly, the eigenfunctions ϕ of a kernel satisfy

$$\int_0^T k(s,t)\phi(t)dt = \lambda\phi(s). \tag{2.17}$$

We say that a kernel is *positive definite* if all its eigenvalues are positive. We say a kernel is symmetric if $k(s,t) = k(t,s)$. Symmetric kernels have real eigenvalues. Symmetric positive definite matrices are of special interest to machine learning because they arise as *covariance matrices* of random vectors. Similarly, symmetric positive-definite kernels arise as *covariance functions* of stochastic processes.

Recall that real symmetric matrices are *diagonalisable*. That is, if A is a real symmetric matrix, there exists an orthogonal transform P such that $P^{-1}AP$ is a diagonal matrix. The columns of P are the eigenvectors of A, which form an orthonormal basis of $\mathbb{R}^N$.

Another way of stating this fact is as follows. If $\{e_i\}$ are the eigenvectors of A and $\{\lambda_i\}$ are the corresponding eigenvalues, then

$$A = \sum_{i=1}^{N} \lambda_i e_i e_i^\top \tag{2.18}$$

Observe that the right-hand side of (2.18) involves the *outer product* of two vectors. The outer product 'combines' two vectors to make a (rank one) matrix. The outer product in $L^2([0,T] : \mathbb{R})$ combines two functions to construct a kernel in a similar manner. Given functions f and g, we can form a kernel by defining $k(s,t) = f(s)g(t)$.

Mercer's theorem states that a continuous symmetric positive-definite kernel function $k(\cdot,\cdot)$ can be 'diagonalised', in much the same way as we diagonalise a symmetric matrix. In order for Mercer's theorem to be applicable, the domain of the function space must be *compact*. Compactness is a generalisation of the intuitive notion of 'finite in extent'. For Euclidean space, it is equivalent to the assumption that the domain is closed and bounded.

Let $\{\phi_i\}$ and $\{\lambda_i\}$ be the eigenfunctions and eigenvalues of k. Then k can be represented in terms of its eigenfunctions as follows:

$$k(s,t) = \sum_{i=1}^{\infty} \lambda_i \phi_i(s)\phi_i(t). \tag{2.19}$$

Mercer's theorem is a special case of one of a large number of results about the eigenvalues of linear operators, collectively known as *spectral theorems*. The term 'spectrum' is used for historical reasons: many of these results were developed by physicists in order to understand the emission spectrum of the hydrogen atom.

To conclude this section, we will present an informal derivation of the basis function expansion of the Dirac delta function. Recall that the Dirac delta function is the 'kernel' defined by

$$\int_0^T \delta(s,t) f(t) dt = f(s) \tag{2.20}$$

for all functions $f \in L^2$. In a sense, *every* function is an eigenfunction of the delta function, just as every vector is an 'eigenvector' of the identity matrix.

As we have mentioned before, it is not strictly permissible to evaluate an element of $L^2(\mathbb{R})$ at a given point, and that such functions should instead be evaluated over a small neighbourhood $[x-\varepsilon, x+\varepsilon]$. In that sense, what follows is correct 'to within an epsilon'.

We begin by fixing s and viewing the delta function as a function of its second argument only. We choose an orthonormal basis $\{\phi_i\}$ and expand $\delta(s,\cdot)$ as in (2.12). We have

$$\begin{aligned}\delta(s,t) &= \sum_{i=1}^{\infty} \langle \delta(s,\cdot), \phi_i \rangle \phi_i(t) \\ &= \sum_{i=1}^{\infty} \left(\int_0^T \delta(s,u)\phi_i(u)du \right) \phi_i(t) \\ &= \sum_{i=1}^{\infty} \phi_i(s)\phi_i(t). \end{aligned} \tag{2.21}$$

This is sometimes known as the 'completeness' property of the basis $\{\phi_i\}$. In the next section, this representation will be instrumental for our understanding of white noise, and hence Brownian motion.

2.1.2 Construction of Brownian motion and series expansions

The random walk interpretation of Brownian motion is not the only useful representation of the process. In later chapters, we will make heavy use of the *Fourier series* construction of Brownian motion. We will demonstrate that Brownian motion has a very simple expression in terms of a generalised Fourier series. To see this, we will first note that there is a relationship between the Dirac delta function and the Brownian covariance function $\min(s,t)$. We first consider the integral of the delta function with respect to its first argument. We have

$$\int_0^t \delta(u,v)du = \mathbb{I}_{\{[0,t]\}}(v). \tag{2.22}$$

To see this, observe that the delta function is identically zero unless v is in the interval $[0,t]$. When v is in this interval, the delta function 'evaluates' the constant function $f(u) = 1$ (which is not written explicitly in the integral on the left). Thus, we conclude that the integral of the delta function is the indicator function on the right.

Integrating both sides of (2.22) with respect to v, we have

$$\int_0^s \int_0^t \delta(u,v)dudv = \int_0^s \mathbb{I}_{\{[0,t]\}}(v)dv$$
$$= \min(s,t). \tag{2.23}$$

We substitute the basis function expansion (2.21) of the delta function into equation (2.23). This gives

$$\min(s,t) = \int_0^s \int_0^t \sum_{i=1}^{\infty} \phi_i(u)\phi_i(v)dudv$$
$$= \sum_{i=1}^{\infty} \left(\int_0^s \phi_i(v)dv\right)\left(\int_0^t \phi_i(u)du\right). \tag{2.24}$$

We now draw an infinite sequence of i.i.d standard normal random variables $\{Z_i\}$, and form the sum

$$W_t = \sum_{i=1}^{\infty} Z_i \int_0^t \phi_i(u)du. \tag{2.25}$$

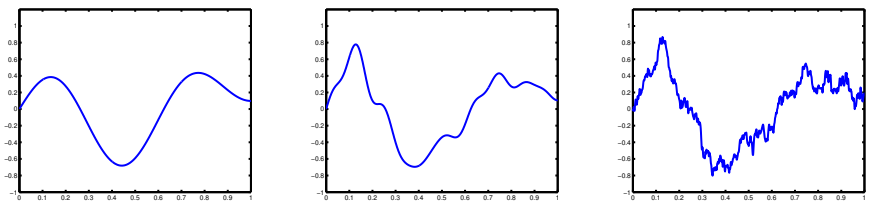


Figure 2.2: Approximate Brownian sample path formed by truncating the series in (2.25) after N terms. We set $N = 10$ (left), $N = 40$ (centre), and $N = 200$ (right). We used the Fourier Sine series (2.10) as an orthonormal basis.

W is a linear combination of (very simple) Gaussian processes, and is therefore a Gaussian process itself. From (2.24), W has covariance function

$$\mathbb{E}[W_sW_t] = \mathbb{E}\left[\left(\sum_{i=1}^{\infty} Z_i \int_0^s \phi_i(u)du\right)\left(\sum_{j=1}^{\infty} Z_j \int_0^t \phi_i(u)du\right)\right]$$

$$= \sum_{i=1}^{\infty}\sum_{j=1}^{\infty} \mathbb{E}[Z_iZ_j]\left(\int_0^s \phi_i(u)du\right)\left(\int_0^t \phi_j(u)du\right) \tag{2.26}$$
$$= \sum_{i=1}^{\infty}\left(\int_0^s \phi_i(u)du\right)\left(\int_0^t \phi_i(u)du\right)$$
$$= \min(s,t). \tag{2.27}$$

so that it is indeed a Brownian motion. Equation (2.25) generalises the well-known *Karhunen-Lòeve* expansion of Brownian motion, in which a Brownian sample path is expanded in terms of the eigenfunctions of $\min(s,t)$.

Equation (2.25) suggests a method for drawing approximate Brownian sample paths. The infinite sum on the right-hand side is truncated after N terms. Figure 2.2 shows how the number of terms in the series affects the approximation. We make use of this approximation in [18], [19] and in Part II of this thesis.

2.2 General theory of nonlinear SDEs

2.2.1 Stochastic differential equations

It is quite difficult to formulate a mathematically consistent continuous-time model of noise that perturbs a dynamical system. In the discrete-time setting, one might introduce a Gaussian white noise – that is, at each time point, the process could be perturbed by an independent draw from a Gaussian distribution. The continuous-time analogue of such a system would satisfy

$$\frac{dx_t}{dt} = a(x_t) + b(x_t)\dot{w}_t, \tag{2.28}$$

where $\dot{w}$ is continuous-time white noise. We allow the amplitude of the noise to depend on the state of the system via a function $b(x)$.

The idea of perturbing the dynamics of a system with white noise does not carry over nicely to the continuous-time setting. This is because $\mathbb{P}(|\dot{w}_t| = \infty) = 1$ at each time t, so it is not clear how to interpret the dynamics of (2.28).

We can overcome this issue by recasting the problem as an *integral equation*. Just as a first-order ordinary differential equation can be written in the form

$$X_t = X_0 + \int_0^t a(X_u)du, \tag{2.29}$$

equation (2.28) can be written in integral form, using a so-called 'Ito integral' as a model of noise. We will explain the precise meaning of the Ito integral term in the next section, but for now it suffices to say that it matches nicely with the naive intuition behind equation (2.28). That is, the system is perturbed by i.i.d noise at every point in time. The integral form of (2.28) is

$$X_t = X_0 + \int_0^t a(X_u)du + \int_0^t b(X_u)dW_u. \tag{2.30}$$

Technically speaking, the term 'stochastic differential equation' is a misnomer, and one should refer to stochastic integral equations instead. However, the following shorthand for (2.30) is commonplace, and serves to sharpen intuition:

$$dX_t = a(X_t)dt + b(X_t)dW_t. \tag{2.31}$$

This can roughly be interpreted as saying 'over a short time Δt, the change in X_t is given by $a(X_t)\Delta t$, perturbed by an independent draw from a Gaussian with mean 0 and variance $b(X_t)^2\Delta t$. Processes of the form (2.31) are referred to as *diffusions*. The Ito model of noise has several appealing properties. Most notably, since the driving Brownian motion W is a Markov process, the diffusion X inherits this property.

2.2.2 Stochastic integration

Central to the theory of diffusion processes is the notion of a *stochastic integral*. Before introducing this idea, we will review some concepts from classical analysis.

Definition 2.2.1. *Given two smooth functions f and g on $[0,T]$ we partition the interval into subintervals of size Δt and form the limit*

$$\begin{aligned}\int_0^T f(u)dg(u) &:= \lim_{\Delta t\to 0}\sum_{n=1}^{T/\Delta t-1} f(n\Delta t)\left(g((n+1)\Delta t)-g(n\Delta t)\right)\\ &= \int_0^T f(u)g'(u)du. \end{aligned}\tag{2.32}$$

We refer to this limit as the Stieltjes integral *of f with respect to g.*

Note that when $g(t) = t$, we recover the usual Riemann integral.

Stieltjes integrals occur naturally in probability theory. If $C(x) = \mathbb{P}(X \le x)$ is the cumulative distribution function of a random variable X, then

$$\mathbb{E}[f(\mathbf{X})] = \int f(x)dC(x). \tag{2.33}$$

This relationship holds even when C has a finite number of jump discontinuitues, for example as in the case of the CDF of the geometric distribution. For CDFs of discrete random variables, we must interpret the derivative of a jump discontinuity as a Dirac delta function. Thus, the theory of Stieltjes integration allows us to unify our treatment of discrete and continuous random variables.

Ideally, we would like to define a Stieltjes integral in which $g = W$, a Brownian motion. However, sample paths of W are nowhere differentiable with probability 1.

This makes the meaning of dW/dt difficult to interpret. The solution is to re-interpret the limit in Definition 2.2.1 as a limit in *mean square*. We say a sequence of random variables $\{X_n\}$ converges to X in mean square if

$$\lim_{n\to\infty} \mathbb{E}\left[(X - X_n)^2\right] = 0. \tag{2.34}$$

To simplify notation, we set $N = T/\Delta t - 1$, leaving dependence on Δt implicit. We can now introduce the Ito integral of a process with respect to Brownian motion.

Definition 2.2.2. *The* Ito integral *of a process X from* 0 *to T with respect to a Brownian motion W is defined as*

$$\int_0^T X_u dW_u := \underset{\Delta t\to 0}{\mathrm{l.i.m}} \sum_{n=0}^{N} X_{n\Delta t}\left(W_{(n+1)\Delta t} - W_{n\Delta t}\right), \tag{2.35}$$

where $\mathrm{l.i.m}$ *denotes 'limit in mean square'.*

A few remarks about this definition are in order. If we want to make sure that the right-hand side of (2.35) converges, we need to be sure that

$$\lim_{\Delta t\to 0} \mathbb{E}\left[\left(\sum_{n=0}^{N} X_{n\Delta t}\left(W_{(n+1)\Delta t} - W_{n\Delta t}\right)\right)^2\right] < \infty. \tag{2.36}$$

Expanding the square, we can see that

$$\begin{aligned}
&\mathbb{E}\left[\left(\sum_{n=0}^{N} X_{n\Delta t}\left(W_{(n+1)\Delta t} - W_{n\Delta t}\right)\right)^2\right] \\
&= \sum_{n=0}^{N}\sum_{m=0}^{N} \mathbb{E}\left[X_{n\Delta t}X_{m\Delta t}\left(W_{(n+1)\Delta t} - W_{n\Delta t}\right)\left(W_{(m+1)\Delta t} - W_{m\Delta t}\right)\right].
\end{aligned} \tag{2.37}$$

If we require that X_{t_1} is independent of the Brownian increment $W_{t_2} - W_{t_1}$ whenever $t_2 > t_1$, then assuming $m > n$, it follows that

$$\begin{aligned}
&\mathbb{E}\left[X_{n\Delta t}X_{m\Delta t}\left(W_{(n+1)\Delta t} - W_{n\Delta t}\right)\left(W_{(m+1)\Delta t} - W_{m\Delta t}\right)\right] \\
&= \mathbb{E}\left[X_{n\Delta t}X_{m\Delta t}\left(W_{(n+1)\Delta t} - W_{n\Delta t}\right)\right]\mathbb{E}\left[\left(W_{(m+1)\Delta t} - W_{m\Delta t}\right)\right] \\
&= 0,
\end{aligned} \tag{2.38}$$

and similarly when $m < n$. Equation (2.38) tells us that most terms in the double sum in (2.37) are 0. We can exploit the independence property to simplify the expression

further:

$$\begin{aligned}
&\sum_{n=0}^{N}\sum_{m=0}^{N}\mathbb{E}\left[X_{n\Delta t}X_{m\Delta t}\left(W_{(n+1)\Delta t}-W_{n\Delta t}\right)\left(W_{(m+1)\Delta t}-W_{m\Delta t}\right)\right]\\
&=\sum_{n=0}^{N}\mathbb{E}\left[X_{n\Delta t}^2\left(W_{(n+1)\Delta t}-W_{n\Delta t}\right)^2\right]\\
&=\sum_{n=0}^{N}\mathbb{E}\left[X_{n\Delta t}^2\right]\mathbb{E}\left[\left(W_{(n+1)\Delta t}-W_{n\Delta t}\right)^2\right]\\
&=\sum_{n=0}^{N}\mathbb{E}[X_{n\Delta t}^2]\Delta t. \qquad (2.39)
\end{aligned}$$

Exchanging the order of expectation and limit, we arrive at

$$\lim_{\Delta t\to 0}\sum_{n=0}^{N}\mathbb{E}[X_{n\Delta t}^2]\Delta t=\mathbb{E}\left[\lim_{\Delta t\to 0}\sum_{n=0}^{N}X_{n\Delta t}^2\Delta t\right]=\mathbb{E}\left[\int_0^T X_u^2\,du\right]. \qquad (2.40)$$

In other words, in order for the limit (2.35) to make sense, our integrand X must be a (possibly random) element of $L^2[0,T]$ with the property that X_{t_1} is always independent of $W_{t_2}-W_{t_1}$ when $t_2>t_1$. If these conditions hold, we say that X is a square integrable, non-anticipative process. It is possible to show that Ito integrals are continuous with probability 1 when viewed as functions of T.

The non-anticipative property of the integrand guarantees that the expectation of an Ito integral is 0. For every choice of Δt,

$$\begin{aligned}
\mathbb{E}\left[\sum_{n=0}^{N}X_{n\Delta t}\left(W_{(n+1)\Delta t}-W_{n\Delta t}\right)\right]&=\sum_{n=0}^{N}\mathbb{E}[X_{n\Delta t}]\mathbb{E}\left[\left(W_{(n+1)\Delta t}-W_{n\Delta t}\right)\right]\\
&=0. \qquad (2.41)
\end{aligned}$$

We have not yet said anything about what happens when the independence assumption used in (2.38) fails. It is possible to relax this assumption, though the theory of stochastic integration becomes vastly more complicated as a result. Anticipative stochastic integrals can be studied by methods such as the so-called *Malliavin calculus* [20], which will not be discussed here.

2.2.3 Ito's lemma

If we recall that the variance of a Brownian increment $\Delta W_t=W_{t+\Delta t}-W_t$ satisfies $\mathrm{Var}[\Delta W_t]=\Delta t$, we can observe that its standard deviation is $\sqrt{\Delta t}$. Therefore, the 'typical size' of a Brownian increment should be of this order. This innocuous-seeming

fact gives stochastic calculus a very different flavour to classical analysis. Suppose we have a smooth function f and a diffusion process X. From Taylor's theorem,

$$f(X_{t+\Delta t}) = f(X_t) + f'(X_t)\Delta X_t + \frac{1}{2}f''(X_t)\Delta X_t^2 + \dots, \tag{2.42}$$

where $\Delta X_t = X_{t+\Delta t} - X_t$. If X were a smooth function, we could discard all but the first two terms on the right hand side, and still expect to have a good estimate of $f(X_{t+\Delta t})$ for small values of Δt. However, since ΔW_t is of order $\sqrt{\Delta t}$, we have

$$\begin{aligned}
\Delta X_t^2 &= (X_{t+\Delta t} - X_t)^2 \\
&= \left(\int_t^{t+\Delta t} a(X_u)du + \int_t^{t+\Delta t} b(X_u)dW_u\right)^2 \\
&\approx (a(X_t)\Delta t + b(X_t)\Delta W_t)^2 \\
&= a(X_t)^2\Delta t^2 + 2a(X_t)b(X_t)\Delta t\Delta W_t + b^2(X_t)\Delta W_t^2,
\end{aligned} \tag{2.43}$$

and the last term on the right is of order Δt. In fact,

$$\mathbb{E}[\Delta W_t^2] = \Delta t \tag{2.44}$$

and

$$\mathrm{Var}[\Delta W_t^2] = 2\Delta t^2, \tag{2.45}$$

so that the variance of ΔW_t^2 is very low for small values of Δt. This suggests that as $\Delta t \to 0$, values taken by ΔW_t^2 cluster around Δt with high probability. We can use this heuristic to produce a new estimate of $f(X_{t+\Delta t})$:

$$\begin{aligned}
f(X_{t+\Delta t}) &\approx f(X_t) + f'(X_t)\Delta X_t + \frac{1}{2}f''(X_t)b^2(X_t)\Delta W_t^2 \\
&\approx f(X_t) + f'(X_t)\left(a(X_t)\Delta t + b(X_t)\Delta W_t\right) + \frac{1}{2}f''(X_t)b^2(X_t)\Delta t.
\end{aligned} \tag{2.46}$$

What we have done here is discard all terms of order greater than Δt in (2.42). If we re-arrange the remaining terms and take a limit as $\Delta t \to 0$, we find that

$$df(X_t) = \left(f'(X_t)a(X_t) + \frac{1}{2}f''(X_t)b^2(X_t)\right)dt + f'(X_t)b(X_t)dW_t, \tag{2.47}$$

or in integral form,

$$f(X_t) = f(X_0) + \int_0^t \left(f'(X_u)a(X_u) + \frac{1}{2}f''(X_u)b^2(X_u)\right)du + \int_0^t f'(X_u)b(X_u)dW_u. \tag{2.48}$$

This is a famous result known as *Ito's lemma*. One consequence of Ito's lemma is that a smooth function of a diffusion is itself a diffusion, as we can see from (2.47).

An analgous result holds for vector-valued diffusions, i.e. when $\mathbf{X}_t \in \mathbb{R}^d$, $\mathbf{W}_t \in \mathbb{R}^d$, $\mathbf{a} : \mathbb{R}^d \to \mathbb{R}^d$ and $\mathbf{b} : \mathbb{R}^d \to \mathbb{R}^{d\times d}$.

There is a time-dependent version of Ito's lemma, which we will need in section 2.2.6. If $f = f(x,t)$, then

$$\begin{aligned} f(X_t,t) =& f(X_0,0) + \int_0^t \left(\dot{f}(X_u,u) + f'(X_u,u)a(X_u) + \frac{1}{2}f''(X_u,u)b^2(X_u) \right) du \\ &+ \int_0^t f'(X_u,u)b(X_u)dW_u, \end{aligned} \tag{2.49}$$

where $\dot{f}$ denotes the partial derivative of f with respect to t, and f' denotes differentiation with respect to x.

2.2.4 The Fokker-Planck equation

We can use Ito's lemma to derive an expression for the *transition density* of a diffusion. This is an important quantity, defined as the function $p(x,t|y,s)$ that satisfies the relation

$$\mathbb{P}(X_t \in A | X_s = y) = \int_A p(x,t|y,s)dx. \tag{2.50}$$

We will often use the shorthand $p(X_t|X_s)$ to denote the transition density.

If X and Y are n-dimensional Euclidean vectors, one way to show $X = Y$ is to prove that $\langle X,Z\rangle = \langle Y,Z\rangle$ for all vectors Z. We will now use an infinite-dimensional equivalent of this argument to derive a relationship between the time and space derivatives of the transition density. This relationship is known as the Fokker-Planck equation. The following argument is adapted from [21], Section 4.3.

Suppose we have an arbitrary smooth function f. Recalling that Ito integrals have mean 0, it follows from (2.48) that

$$\mathbb{E}[f(X_t)] = \mathbb{E}\left[f(X_0) + \int_0^t \left(f'(X_u)a(X_u) + \frac{1}{2}f''(X_u)b^2(X_u) \right) du \right]. \tag{2.51}$$

If we assume f is sufficiently well-behaved, we can exchange differentiation and expectation. Differentiating (2.51) with respect to time, we find that

$$\frac{d}{dt}\mathbb{E}[f(X_t)] = \mathbb{E}\left[f'(X_t)a(X_t) + \frac{1}{2}f''(X_t)b^2(X_t) \right]. \tag{2.52}$$

Now, setting $X_0 = x_0$, we can use the definition of $\mathbb{E}[\,\cdot\,]$ to rewrite (2.52) as

$$\int f(x)\frac{\partial}{\partial t}p(x,t|x_0,0)dx = \int \left(f'(x)a(x) + \frac{1}{2}f''(x)b^2(x) \right) p(x,t|x_0,0)dx, \tag{2.53}$$

where the integral is over all nonzero values of $p(x,t|x_0,0)$. If we impose some natural decay assumptions on $p(x,t|x_0,0)$ (so that, for instance, $p(x,t|x_0,0) \to 0$ as $x \to \infty$) , we can integrate the right-hand side by parts:

$$\begin{aligned}&\int f(x)\frac{\partial}{\partial t}p(x,t|x_0,0)dx\\ =&\int f(x)\left(-\frac{\partial}{\partial x}\left(a(x)p(x,t|x_0,0)\right)+\frac{1}{2}\frac{\partial^2}{\partial x^2}\left(b^2(x)p(x,t|x_0,0)\right)\right)dx.\end{aligned} \tag{2.54}$$

Since this expression holds for all smooth f, we can conclude [1] that the transition density satisfies

$$\frac{\partial}{\partial t}p(x,t|x_0,0) = -\frac{\partial}{\partial x}\left(a(x)p(x,t|x_0,0)\right)+\frac{1}{2}\frac{\partial^2}{\partial x^2}\left(b^2(x)p(x,t|x_0,0)\right). \tag{2.55}$$

The argument can be adapted to work for a multidimensional diffusion process $\mathbf{X}_t$ satisfying the SDE

$$d\mathbf{X}_t = \mathbf{a}(\mathbf{X}_t)dt + \mathbf{B}(\mathbf{X}_t)d\mathbf{W}_t. \tag{2.56}$$

The transition density $p(\mathbf{x},t|\mathbf{y},s)$ of $\mathbf{X}$ satisfies the multidimensional Fokker-Planck equation

$$\frac{\partial}{\partial t}p(\mathbf{x},t|\mathbf{y},s) = -\sum_{i=1}^{d}\frac{\partial}{\partial x_i}\left(\mathbf{a}(\mathbf{x})p(\mathbf{x},t|\mathbf{y},s)\right)+\frac{1}{2}\sum_{i=1}^{d}\sum_{j=1}^{d}\frac{\partial^2}{\partial x_i\partial x_j}\left(\mathbf{B}(\mathbf{x})\mathbf{B}^{\mathsf{T}}(\mathbf{x})p(\mathbf{x},t|\mathbf{y},s)\right). \tag{2.57}$$

If a process starts at x_0 at time 0 and we want to know its probability density at time t, we must solve this PDE. The solution can be computed in closed form only in a few very special cases, such as when **a** and **b** are linear. In general, one must resort to numerical methods to find a solution. In fact, when $d > 3$, the situation is typically reversed. If one has a PDE of the form (2.57), it is often the case that the only practical way to solve it is via a Monte-Carlo method based on diffusion processes.

We will finish this section with a short digression on the intuitive interpretation of Equation (2.55). Given the dynamics of an ODE

$$\frac{dy_t}{dt} = f(y_t), \tag{2.58}$$

one can often hope to understand some properties of the dynamics intuitively, 'by inspection'. The intuition behind equation (2.58) is roughly as follows: *starting from time t, the change in y over a short time Δt is $f(y_t)\Delta t$.*

[1] The reason we can draw this conclusion is that the linear span of smooth functions with compact support is dense in $L^2(\mathbb{R}^d)$. The argument is analogous to proving $X = Y$ by showing $\forall Z, \langle X,Z\rangle = \langle Y,Z\rangle$.

Suppose we take the simplest case of (2.55), setting the drift function a to 0 and the diffusion coefficient to $b(x) = 1$ (and ignoring the initial conditions), so that

$$\frac{\partial}{\partial t} p(x,t) = \frac{1}{2} \frac{\partial^2}{\partial x^2} p(x,t). \tag{2.59}$$

It is not immediately clear that one can find the same kind of intuition as in (2.58). However, things become clearer if we 'take a step back' from the limit in the definition of the second derivative on the right-hand side. Instead of (2.59), fix a location x_0 in space and consider $p(x_0,t)$ as a function of time only. We have

$$\frac{\partial}{\partial t} p(x_0,t) \approx \frac{1}{2} \frac{p(x_0+\varepsilon,t) + p(x_0-\varepsilon,t) - 2p(x_0,t)}{\varepsilon^2}, \tag{2.60}$$

which is equal to (2.59) in the limit as $\varepsilon \to 0$. We are now *almost* in a position to apply the same intuition to (2.60) as we did to (2.58). However, the dynamics of $p(x_0,t)$ are now given in terms of two *new* unknown functions, $p(x_0+\varepsilon,t)$ and $p(x_0-\varepsilon,t)$. These, too, depend only on time. We can repeat the analysis above with $x_0 \pm \varepsilon$ in place of x_0. The conclusion is that the functions $\{p(x_0,t), p(x_0 \pm \varepsilon,t), p(x_0 \pm 2\varepsilon,t), \dots\}$ form an *infinite system of ordinary differential equations*.

For an ODE, given the state of the system at some initial time t_0, we can determine the behaviour of the system at any time t. However, for a PDE we need some extra information. In order to solve the system of ODEs, we need to know the behaviour of the system on some *boundary*. In the one-dimensional case, this means we need to know two functions: $p(b^+,t)$ and $p(b^-,t)$, with $b^- < x < b^+$. We can then determine the behaviour of $\{p(b^+-\varepsilon,t), p(b^+-2\varepsilon,t), \dots\}$ and $\{p(b^-+\varepsilon,t), p(b^-+2\varepsilon,t), \dots\}$.

Reasoning such as this is the basis behind the simplest family of numerical methods for PDEs. These methods are known as *finite difference methods*.

2.2.5 The Euler-Maruyama approximation

Diffusion processes are infinite-dimensional objects. As such, some form of discretisation is necessary if one wants to simulate a diffusion sample path on a computer. We will now discuss the simplest form of discretisation scheme – the *Euler-Maruyama approximation* [22].

The scheme is derived by observing that

$$\mathbf{X}_{(k+1)\Delta t} - \mathbf{X}_{k\Delta t} = \int_{k\Delta t}^{(k+1)\Delta t} \mathbf{a}(\mathbf{X}_u)du + \int_{k\Delta t}^{(k+1)\Delta t} \mathbf{b}(\mathbf{X}_u)d\mathbf{W}_u, \tag{2.61}$$

and making the approximations

$$\int_{k\Delta t}^{(k+1)\Delta t} \mathbf{a}(\mathbf{X}_u)du \approx \mathbf{a}(\mathbf{X}_{k\Delta t})\Delta t, \tag{2.62}$$

and

$$\begin{aligned}\int_{k\Delta t}^{(k+1)\Delta t} \mathbf{b}(\mathbf{X}_u)d\mathbf{W}_u &\approx \mathbf{b}(\mathbf{X}_{k\Delta t}) \int_{k\Delta t}^{t+\Delta t} d\mathbf{W}_u \\ &= \mathbf{b}(\mathbf{X}_{k\Delta t}) \left(\mathbf{W}_{(k+1)\Delta t} - \mathbf{W}_{k\Delta t}\right).\end{aligned} \tag{2.63}$$

The interval $[0,T]$ is divided into N timesteps of length Δt, and the diffusion is approximated using the following recursive construction:

$$\mathbf{X}_{(k+1)\Delta t} = \mathbf{X}_{k\Delta t} + \mathbf{a}(\mathbf{X}_{k\Delta t})\Delta t + \mathbf{b}(\mathbf{X}_{k\Delta t})\sqrt{\Delta t}\mathbf{Z}_k, \tag{2.64}$$

where $\{\mathbf{Z}_i\}$ are a set of independent standard d-dimensional Gaussians. Values of the diffusion at other times t are computed by linear interpolation. This approximation makes it clear that over small timescales the transition density $p(\,\cdot\,,(k+1)\Delta t|\mathbf{X}_{k\Delta t},k\Delta t)$ behaves like a Gaussian density with mean $\mathbf{a}(\mathbf{X}_{k\Delta t})\Delta t$ and variance $\mathbf{b}\mathbf{b}^\intercal(\mathbf{X}_{k\Delta t})\Delta t$ (we have used a centered dot to emphasize that we are thinking of p as a function of its first argument only).

2.2.6 Conditioned diffusions

The Euler-Maruyama approximation gives us a means of generating sample paths of a diffusion process. However, this may not be sufficient for our purposes. If our process has the initial value x_0 at time 0 and we observe that it takes the value x_T at time T, we might want to simulate sample paths that are consistent with this observation. To this end, we would like to know the dynamics of the conditioned process $X|\{X_0 = x_0,\, X_T = x_T\}$, which we refer to as a *diffusion bridge*. It turns out that diffusion bridges are also diffusions so that, somewhat counterintuitively, they have the Markov property.

Perhaps the simplest example of a diffusion bridge is the process obtained by conditioning a Brownian motion to satisfy $W_1 = 0$. The resulting process is known as the *Brownian bridge*. The brownian bridge is a Gaussian process with mean 0 and covariance $k(s,t) = \min(s,t) - st$, $s,t \in [0,1]$. As we wil see, the dynamics of the Brownian bridge can be written in the form of a SDE, with dynamics

$$dX_t = -\frac{X_t}{1-t} + dW_t. \tag{2.65}$$

It is surprisingly difficult to find a general treatment of diffusion bridges in the literature. A terse and rather abstract account can be found in [15] as a special case of the more general *Doob h-transform*. We now present a derivation of the dynamics of a diffusion bridge. The derivation is adapted from [23].

We will modify the argument used in section 2.2.4 to derive a Fokker-Planck equation for a one-dimensional diffusion bridge. The extension to multiple dimensions is similar. Define the conditional density $p(x,t|x_0,0,y,T)$ as the function that satisfies

$$\mathbb{P}(X_t \in A|X_0 = x_0,\ X_T = y) = \int_A p(x,t|x_0,0,y,T)dx. \tag{2.66}$$

It follows from the Markov property that one can rewrite this density as

$$p(x,t|x_0,0,y,T) = \frac{p(y,T|x,t)p(x,t|x_0,0)}{p(y,T|x_0,0)}, \tag{2.67}$$

where the terms on the right are unconditional transition densities. Thus, if f is a well-behaved function, it follows that

$$\begin{aligned}\mathbb{E}\left[f(X_t)|X_0 = x_0, X_T = y\right] &= \int f(x)p(x,t|x_0,0,y,T)dx \\ &= \int f(x)\frac{p(y,T|x,t)p(x,t|x_0,0)}{p(y,T|x_0,0)}dx \\ &= \frac{\mathbb{E}[f(X_t)p(y,T|X_t,t)|X_0 = x_0]}{p(y,T|x_0,0)}.\end{aligned} \tag{2.68}$$

The expectation on the right-hand side is not conditioned on a future time, so we can apply the time-dependent version of Ito's lemma (2.49) to $g(x,t) = f(x)p(y,T|x,t)$ as usual. We can re-write the numerator of the right-hand side as

$$\begin{aligned}&\mathbb{E}[g(X_t,t)|X_0 = x_0] \\ &= \mathbb{E}\left[\int_0^t \left(a\frac{\partial}{\partial x}g(X_s,s) + \frac{1}{2}b^2\frac{\partial^2}{\partial x^2}g(X_s,s) + \frac{\partial}{\partial t}g(X_s,s)\right)ds \mid X_0 = x_0\right].\end{aligned} \tag{2.69}$$

Now, since $p(y,T|x,t)$ is the density of a diffusion process, one can show that it satisfies the *backward Kolmogorov equation*[2] (which we do not derive here):

$$a(x)\frac{\partial}{\partial x}p + \frac{1}{2}b^2(x)\frac{\partial^2}{\partial x^2}p + \frac{\partial}{\partial t}p = 0. \tag{2.70}$$

[2]The spatial differential operator that appears in the backward Kolmogov equation is the formal adjoint on $L^2(\mathbb{R})$ of the spatial differential operator that appears in the Fokker-Planck equation. The PDEs are closely related.

If we expand (2.69), and apply the backward Kolmogorov equation to cancel terms, we are left with

$$\begin{aligned}&\mathbb{E}[f(X_t)p(y,T|X_t,t)|X_0=x_0]\\&=\mathbb{E}\left[\int_0^t\left(ap\frac{\partial}{\partial x}f+\frac{1}{2}b^2p\frac{\partial^2}{\partial x^2}f+b^2\frac{\partial}{\partial x}p\frac{\partial}{\partial x}f\right)ds\mid X_0=x_0\right].\end{aligned}\tag{2.71}$$

Note that we supressed the function arguments in this expression for brevity. We can re-arrange the expression and substitute it into equation (2.68), which shows that

$$\begin{aligned}&\mathbb{E}[f(X_t)|X_0=x_0,\ X_T=y]\\&=\frac{1}{p(y,T|x_0,0)}\mathbb{E}\left[\int_0^t\left(\left(ap+b^2\frac{\partial}{\partial x}p\right)\frac{\partial}{\partial x}f+\frac{1}{2}b^2p\frac{\partial^2}{\partial x^2}f\right)\mid X_0=x_0\right].\end{aligned}$$

Differentiating both sides with respect to t and converting the right-hand side back to a conditional probability using (2.67) (dividing above and below by $p(y,T|x,t)$ where necessary), we have

$$\begin{aligned}&\frac{\partial}{\partial t}\mathbb{E}[f(X_t)|X_0=x_0,\ X_T=y]\\&=\mathbb{E}\left[\left(a+\frac{b^2}{p}\frac{\partial}{\partial x}p\right)\frac{\partial}{\partial x}f+\frac{1}{2}b^2\frac{\partial^2}{\partial x^2}f\mid X_0=x_0,X_T=y\right]\\&=\mathbb{E}\left[\left(a+b^2\frac{\partial}{\partial x}\log(p)\right)\frac{\partial}{\partial x}f+\frac{1}{2}b^2\frac{\partial^2}{\partial x^2}f\mid X_0=x_0,X_T=y\right].\end{aligned}\tag{2.72}$$

If we set $\hat{a}=a+b^2\partial\log(p)/\partial x$, we can use integration by parts as in section 2.2.4 to deduce the Fokker-Planck equation. We see that the diffusion bridge satisfies the following SDE:

$$dX_t=\left(a(X_t)+b^2(X_t)\frac{\partial}{\partial x}\log\left(p(y,T|X_t,t)\right)\right)dt+b(X_t)dW_t.\tag{2.73}$$

In other words, we can condition the diffusion to hit y at time T by modifying the drift function. However, this is typically difficult to do in practice, since one must first calculate the transition density $p(y,T|x,t)$ for all values of x and t.

2.2.7 Girsanov's theorm in practice

In this section we will briefly review the concepts undelying importance sampling, and show how they can be applied to Brownian motion and related processes. The main result that we describe in this section is known as Girsanov's theorem. The rigorous statement of Girsanov's theorem involves technical measure-theoretic concepts such as

absolute continuity of measures and the Radon-Nikodym derivative. A full account of the result can be found in any good textbook on stochastic analysis - see, for example, [9] [10]. Rather than reproduce an already well-known result, the aim of this section is to describe Girsanov's theorem in a way that is as accessible as possible. To this end, we first review the ideas behind importance sampling, and show how they generalise to diffusion processes.

Suppose we have a random variable X with density p_X. By definition, the expectation of $f(X)$ is

$$\mathbb{E}[f(X)] = \int f(x)p_X(x)dx. \tag{2.74}$$

Now suppose we have some other random variable Y with density p_Y. If the support of p_X is contained in the support of p_Y so that $p_Y(x) = 0 \implies p_X(x) = 0$, we can rewrite the expectation in terms of Y as follows:

$$\begin{aligned}\int f(x)p_X(x)dx &= \int f(x)\left(\frac{p_X(x)}{p_Y(x)}\right)p_Y(x)dx \\ &= \mathbb{E}\left[f(Y)\left(\frac{p_X(Y)}{p_Y(Y)}\right)\right].\end{aligned} \tag{2.75}$$

One can now estimate $\mathbb{E}[f(X)]$ by drawing samples from p_Y and computing a weighted average. If $Y^{(i)}$ is a sample from p_Y, we define the corresponding importance weight to be

$$w^{(i)} = \frac{p_X\left(Y^{(i)}\right)}{p_Y\left(Y^{(i)}\right)}. \tag{2.76}$$

Our estimate of $\mathbb{E}[f(X)]$ is then

$$\mathbb{E}[f(X)] \approx \frac{1}{N}\sum_{i=1}^{N} f\left(Y^{(i)}\right)w^{(i)}. \tag{2.77}$$

This trick is useful in when we do not know how to sample from X directly but can evaluate its density. It can also be exploited to reduce the variance of a Monte-Carlo estimate of $\mathbb{E}[f(X)]$. For example, if X is a standard normal random variable and we attempt to compute $\mathbb{P}(X > 5)$, we will typically need to draw over a million samples to obtain a single sample satisfying $X^{(i)} > 5$. One could use importance sampling to draw samples from a normal distribution with mean 5 instead.

Girsanov's theorem says that an infinite-dimensional version of importance sampling can be applied to a diffusion process. Suppose that we wish to calculate $\mathbb{E}[f(X_T)]$ for a given function f.

One could draw a large number of sample paths $X^{(i)}$, and compute the mean value:

$$\mathbb{E}[f(X_T)] \approx \frac{1}{N}\sum_{i=1}^{N} f\left(X_T^{(i)}\right). \tag{2.78}$$

One way of drawing sample paths is to use the Euler-Maruyama approximation. The interval $[0,T]$ is divided up into $n = T/\Delta t$ timesteps, and X is approximated via the recursion

$$X_{(k+1)\Delta t} = X_{k\Delta t} + a(X_{k\Delta t})\Delta t + b(X_{k\Delta t})Z_k\sqrt{\Delta t}, \tag{2.79}$$

where $\{Z_i\}$ are i.i.d standard normal random variables. In this sense, we can write

$$X_T = g(X_0, Z_1, \ldots, Z_n). \tag{2.80}$$

That is, X_T is completely determined by X_0 and the values of the random variables $\{Z_i\}$ that appear in (2.79). Here, g is defined implicitly in the Euler-Maruyama recursion.

Instead of drawing i.i.d standard normal random variables $\{Z_i\}$, we can draw importance variates from a normal distribution with an altered mean, which possibly depends on the state of $X_{k\Delta t}$. We set

$$V_k = Z_k + u(X_{k\Delta t})\sqrt{\Delta t}. \tag{2.81}$$

We use the random variables $\{V_i\}$ in place of $\{Z_i\}$ in (2.79), which gives a new process $\bar{X}$ defined by

$$\begin{aligned}\bar{X}_{(k+1)\Delta t} &= \bar{X}_{k\Delta t} + a(\bar{X}_{k\Delta t})\Delta t + b(\bar{X}_{k\Delta t})V_k\sqrt{\Delta t},\\ &= \bar{X}_{k\Delta t} + \left(a(\bar{X}_{k\Delta t}) + u(\bar{X}_{k\Delta t})b(\bar{X}_{k\Delta t})\right)\Delta t + b(\bar{X}_{k\Delta t})Z_k\sqrt{\Delta t}.\end{aligned} \tag{2.82}$$

By modifying the mean of the variates $\{Z_i\}$, we have changed the drift of the process X. We can generate sample paths from $\bar{X}$ to estimate the expectation in (2.78), but we must compensate for the fact that we sampled from $\{V_i\}$ rather than $\{Z_i\}$ (note that we now modify the distribution of V dependent on the state of $\bar{X}$ rather than the state of X as in equation (2.81)). The importance ratio is given by

$$\begin{aligned}\frac{p_Z(V_{1:n})}{p_V(V_{1:n})} &= \exp\left(-\sum_{i=1}^{n}\frac{V_i^2}{2}\right) / \exp\left(-\sum_{i=1}^{n}\frac{\left(V_i - u(\bar{X}_{i\Delta t})\sqrt{\Delta t}\right)^2}{2}\right)\\ &= \exp\left(-\sum_{i=1}^{n}\frac{\left(Z_i + u(\bar{X}_{i\Delta t})\sqrt{\Delta t}\right)^2}{2}\right) / \exp\left(-\sum_{i=1}^{n}\frac{Z_i^2}{2}\right)\\ &= \exp\left(\sum_{i=1}^{n} u(\bar{X}_{i\Delta t})Z_i\sqrt{\Delta t} - \frac{1}{2}\sum_{i=1}^{n} u^2(\bar{X}_{i\Delta t})\Delta t\right).\end{aligned} \tag{2.83}$$

Observe that for small values of Δt, the first term in the exponential approximates a stochastic integral and the second term approximates a classical integral. As $\Delta t \to 0$, the importance ratio converges to

$$w = \exp\left(\int_0^T u(\bar{X}_s)dW_s - \frac{1}{2}\int_0^T u^2(\bar{X}_s)ds\right), \tag{2.84}$$

and the process $\bar{X}$ converges to the diffusion

$$d\bar{X}_t = (a(\bar{X}_t) + u(\bar{X}_t)b(\bar{X}_t))\,dt + b(\bar{X}_t)dW_t. \tag{2.85}$$

Thus, in order to estimate (2.78), we can draw samples from (2.85) and weight them by (2.84):

$$\mathbb{E}[f(X_T)] \approx \frac{1}{N}\sum_{i=1}^{N} f\left(\bar{X}_T^{(i)}\right)\exp\left(\int_0^T u\left(\bar{X}_s^{(i)}\right)dW_s^{(i)} - \frac{1}{2}\int_0^T u^2\left(\bar{X}_s^{(i)}\right)ds\right). \tag{2.86}$$

Unless each term in (2.86) can be computed analytically, one must sample approximate paths using (2.82) and approximate importance weights using (2.83).

The density ratio (2.84) has applications beyond importance sampling. For example, one can take the expectation of the log of this quantity to define a notion of Kullback-Liebler divergence between two diffusions. This approach was used to good effect in [24].

Note that we cannot modify the variance of $\{Z_i\}$ in the same way that we modified the mean. If we attempt to modify the variance of the random variables $\{Z_i\}$ *en masse*, then the terms involving Z_i^2 in (2.83) will not cancel, and the ratio will tend to either 0 or infinity as $\Delta t \to 0$. On the other hand, if we change the distribution of *one* variable from the set $\{Z_i\}$, the contribution from that random variable tends to 0 in the limit.

In Chapter 8, we suggest an alternative parametrisation of the Brownian motion driving the SDE. We apply importance sampling to the Fourier series coefficients of the Brownian motion (See Section 2.1.2), which also happen to be standard normal variates. This parametrisation allows us to modify the distribution of a small number of variates without the modification disappearing in the limit as $\Delta t \to 0$. As a result, we have more scope to construct interesting importance distributions. It is possible that the Fourier series methodology is mathematically equivalent to Girsanov's theorem (though we have neither proved nor disproved this assertion). However, we argue that the Fourier series approach provides insights that are not obvious using the traditional Girsanov drift-modification approach.

Chapter 3

The filtering problem

In this chapter, we will review the 'filtering problem'. That is, given a sequence of noisy observations of a stochastic process, how can one 'filter' out the noise while observing the process in real time? We begin with a review of the Kalman filter, which is the de-facto standard method of solving filtering problems. Having introduced the Kalman filter and its nonlinear counterpart (the 'extended' Kalman filter), we will expand our discussion by reviewing some more efficient filters. In Section 3.1.2 we introduce the 'unscented' filter, and in Section 3.3 we describe the particle filter.

3.1 The Kalman filter

The Kalman filter was developed as an extension of least-squares estimation to linear dynamical systems. The aim of the filter is to compute the distribution of a signal $\mathbf{X}$ at a sequence of times $\{t_1, \ldots t_n\}$ based on a series of noisy observations $\{\mathbf{Y}_{t_i}\}$ up to and including those times.

Prior to Kalman's paper [25], the *Wiener-Kolmogorov* filter was the standard tool for linear time series estimation problems. The idea behind the Wiener-Kolmogorov filter is to examine the time series of interest in the frequency domain, and remove the high-frequency components (i.e. noise) in a principled manner. However, the method required advanced knowledge of Fourier analysis, and was not easily digestible by engineering undergraduates. The Kalman filter simplified this methodology considerably, shifting the emphasis from a 'function space' interpretation of the filtering problem to a 'state space' interpretation. It is worth noting that the Kalman and Wiener-Kolmogorov filters make identical predictions in finite-dimensional linear filtering problems [26]. An alternative formulation of the Kalman filter was developed by Swerling [27], and

was published the year before Kalman's paper.

The success of the Kalman filter is also attributable to the fact that it is computationally efficient and recursive, which means it is easily implementable on a computer. Both Kalman's and Swerling's papers were coincident with the rise of large-scale scientific computing. One can make the case that the success of the Kalman filter is attributable to the rise of the computer.

There is a vast literature that builds on Kalman's paper [25], and the filter has had many high-profile applications. Perhaps most notably, it was used as part of the navigation system in the Apollo program [28]. Beyond applications in target tracking and navigation, the Kalman filter has found applications in such diverse fields as econometrics, geostatistics, and robotics. We now present a short, informal derivation of the linear filter before discussing its extension to nonlinear system models.

The system is assumed to be linear Gaussian, so that $\mathbf{X}_{t_0} \in \mathbb{R}^n$ follows a Gaussian distribution, and the dynamics satisfy

$$\mathbf{X}_{t_k} = \mathbf{A}\mathbf{X}_{t_{k-1}} + \mathbf{Z}_{t_{k-1}} + \mathbf{c}. \tag{3.1}$$

Here, $\{\mathbf{Z}_{t_i}\}$ represents n-dimensional discrete-time Gaussian noise noise with covariance $\mathbf{Q}$, $\mathbf{A} \in \mathbb{R}^{n\times n}$, and $\mathbf{c} \in \mathbb{R}^n$.

We assume that observations $\{\mathbf{Y}_{t_i}\}$ are made in the form

$$\mathbf{Y}_{t_k} = \mathbf{H}\mathbf{X}_{t_k} + \mathbf{V}_{t_k}, \tag{3.2}$$

where $\mathbf{Y}_{t_k} \in \mathbb{R}^s$, $\mathbf{H} \in \mathbb{R}^{s\times n}$, and $\{\mathbf{V}_{t_i}\}$ are i.i.d Gaussian with covariance $\mathbf{R}$.

The state transition equation (3.1) and the observation equation (3.2) are *linear*. This ensures that $\{\mathbf{X}_{t_1}, \mathbf{Y}_{t_1}, \mathbf{X}_{t_2}, \dots\}$ are jointly Gaussian. The multivariate Gaussian distribution is closed under conditioning, so that the filtering distribution $p(\mathbf{X}_{t_k}|\mathbf{Y}_{t_{1:k}})$ is also Gaussian.

The filter is divided into two steps. Starting at time t_{k-1}, we first apply the *prediction step*, where the known dynamics of the system are used to compute the distribution of the state at time t_k. That is, we aim to compute the mean and covariance of $p(\mathbf{X}_{t_k}|\mathbf{Y}_{t_{1:k-1}})$. In the update step, we make a new observation, and incorporate that information into our estimate to arrive at the time-t_k filtering distribution $p(\mathbf{X}_{t_k}|\mathbf{Y}_{t_{1:k}})$.

Let $\mathbf{m}_{t_{k-1}}$ and $\mathbf{P}_{t_{k-1}}$ represent the mean and covariance of the filtering distribution at time t_{k-1}, and suppose $\bar{\mathbf{X}}_{t_{k-1}} \sim \mathcal{N}(\mathbf{m}_{t_{k-1}}, \mathbf{P}_{t_{k-1}})$. That is, $\bar{\mathbf{X}}_{t_{k-1}}$ represents some 'plausible' realisation of the location of the signal, given our current knowledge about its location.

In the prediction step, we compute the effect of applying the transition equation (3.1) to $\bar{\mathbf{X}}_{t_{k-1}}$. We find that the instant before the observation $\mathbf{Y}_{t_k}$ arrives, the filtering mean and covariance satisfy

$$\mathbf{m}_{t_k}^- := \mathbb{E}\left[\mathbf{A}\bar{\mathbf{X}}_{t_{k-1}} + \mathbf{Z}_{t_{k-1}} + \mathbf{c}\right] = \mathbf{A}\mathbf{m}_{t_{k-1}} + \mathbf{c}, \tag{3.3}$$

and

$$\mathbf{P}_{t_k}^- := \mathrm{Cov}\left(\mathbf{A}\bar{\mathbf{X}}_{t_{k-1}} + \mathbf{Z}_{t_{k-1}}\right) = \mathbf{A}\mathbf{P}_{t_{k-1}}\mathbf{A}^\top + \mathbf{Q}. \tag{3.4}$$

For the update step, we must incorporate the information from the observation $\mathbf{Y}_{t_k}$. We first compute the predicted values of the observation. Suppose $\mathbf{X}_{t_k}^- \sim \mathcal{N}(\mathbf{m}_{t_k}^-, \mathbf{P}_{t_k}^-)$. We have

$$\mu_{t_k} := \mathbb{E}\left[\mathbf{H}\mathbf{X}_{t_k}^- + \mathbf{V}_{t_k}\right] = \mathbf{H}\mathbf{m}_{t_k}^-, \tag{3.5}$$

$$\mathbf{S}_{t_k} := \mathrm{Cov}\left(\mathbf{H}\mathbf{X}_{t_k}^- + \mathbf{V}_{t_k}\right) = \mathbf{H}\mathbf{P}_{t_k}^-\mathbf{H}^\top + \mathbf{R} \tag{3.6}$$

$$\mathbf{C}_{t_k} := \mathrm{Cov}\left(\mathbf{H}\mathbf{X}_{t_k}^- + \mathbf{V}_{t_k}, \mathbf{X}_{t_k}^-\right) = \mathbf{H}\mathbf{P}_{t_k}^- \tag{3.7}$$

We can now form the joint distribution of the predicted state and observation, concluding that

$$\mathbb{E}[\mathbf{X}_{t_k}, \mathbf{Y}_{t_k} | \mathbf{Y}_{t_{1:k-1}}] = (\mathbf{m}_{t_k}^-, \mu_{t_k}) \tag{3.8}$$

and

$$\mathrm{Cov}(\mathbf{X}_{t_k}, \mathbf{Y}_{t_k} | \mathbf{Y}_{t_{1:k-1}}) = \begin{pmatrix} \mathbf{P}_{t_k}^- & \mathbf{C}_{t_k}^\top \\ \mathbf{C}_{t_k} & \mathbf{S}_{t_k} \end{pmatrix}. \tag{3.9}$$

Standard results about the multivariate Gaussian distribution allow us to conclude that the posterior mean and covariance satisfy

$$\mathbb{E}[\mathbf{X}_{t_k} | \mathbf{Y}_{t_{1:k-1}}, \mathbf{Y}_{t_k}] = \mathbf{m}_{t_k}^- + \mathbf{C}_{t_k}\mathbf{S}_{t_k}^{-1}(\mathbf{Y}_{t_k} - \mu_{t_k}) \tag{3.10}$$

and

$$\mathrm{Cov}(\mathbf{X}_{t_k} | \mathbf{Y}_{t_{1:k-1}}, \mathbf{Y}_{t_k}) = \mathbf{P}_{t_k}^- - \mathbf{C}_{t_k}\mathbf{S}_{t_k}^{-1}\mathbf{C}_{t_k}^\top. \tag{3.11}$$

Note that the assumption of Gaussian errors is not strictly necessary, though it makes the derivation more straightforward. There are non-Bayesian derivations of the Kalman filter (such as Kalman's original derivation) that show that among all linear filters, the Kalman filter minimises the error in the mean square sense, regardless of the nature of the observation and process noise.

The quantity $\mathbf{K}_{t_k} = \mathbf{C}_{t_k}\mathbf{S}_{t_k}^{-1}$ is known as the *Kalman gain*, and equations (3.10) and (3.11) can be written as

$$\mathbb{E}[\mathbf{X}_{t_k} | \mathbf{Y}_{t_{1:k-1}}, \mathbf{Y}_{t_k}] = \mathbf{m}_{t_k}^- + \mathbf{K}_{t_k}(\mathbf{Y}_{t_k} - \mu_{t_k}) \tag{3.12}$$

and

$$\mathrm{Cov}(\mathbf{X}_{t_k}|\mathbf{Y}_{t_{1:k-1}},\mathbf{Y}_{t_k}) = \mathbf{P}^-_{t_k} - \mathbf{K}_{t_k}\mathbf{S}_{t_k}\mathbf{K}^\top_{t_k} \tag{3.13}$$

When attempting to construct a filter, one has two sources of information: prior knowledge of the system dynamics, and noisy observations. One must therefore look for a method of combining these sources of information. The Kalman gain can be understood as a kind of coefficient that tells us how to combine information from the observations with our prior knowledge of the dynamics. From (3.7) and (3.6), we have

$$\mathbf{K}_{t_k} = \mathbf{H}\mathbf{P}^-_{t_k}\left(\mathbf{H}\mathbf{P}^-_{t_k}\mathbf{H}^\top + \mathbf{R}\right)^{-1}, \tag{3.14}$$

The intuition behind the Kalman gain is perhaps best explained in one dimension, so that $\mathbf{R}$, $\mathbf{H}$ and $\mathbf{P}^-_{t_k}$ are real numbers. When $\mathbf{R}$ is large compared to $\mathbf{H}$ and $\mathbf{P}^-_{t_k}$, we have $\mathbf{K} \approx 0$. Equations (3.12) and (3.13) then tell us that we make essentially no modification to our posterior mean and covariance. But $\mathbf{R}$ is large exactly when our observations are imprecise. Thus, when our observations are noisy, we rely primarily on our knowledge of the signal dynamics.

On the other hand, when $\mathbf{R}$ is small, we make very precise observations. In this case,

$$\mathbf{K}_{t_k} \approx 1/\mathbf{H} \tag{3.15}$$

and

$$\mathbf{Y}_{t_k} \approx \mathbf{H}\mathbf{X}_{t_k}. \tag{3.16}$$

It follows from the definition (3.5) of μ_{t_k} that

$$\mathbf{K}_{t_k}\mu_{t_k} = \mathbf{K}_{t_k}\mathbf{H}\mathbf{m}^-_{t_k} \approx \mathbf{m}^-_{t_k}. \tag{3.17}$$

Thus, the posterior mean (3.12) satisfies

$$\begin{aligned}\mathbf{m}_{t_k} &= \mathbf{m}^-_{t_k} + \mathbf{K}_{t_k}(\mathbf{Y}_{t_k} - \mu_{t_k}) \\ &\approx \left(\mathbf{m}^-_{t_k} - \mathbf{K}_{t_k}\mu_{t_k}\right) + \mathbf{K}_{t_k}\mathbf{H}\mathbf{X}_{t_k} \\ &\approx \mathbf{X}_{t_k}.\end{aligned} \tag{3.18}$$

For the posterior covariance, we use the approximation (3.15) along with the definition (3.6) of $\mathbf{S}_{t_k}$ to see that

$$\begin{aligned}\mathbf{P}_{t_k} &= \mathbf{P}^-_{t_k} - \mathbf{K}_{t_k}\mathbf{S}_{t_k}\mathbf{K}_{t_k} \\ &= \mathbf{P}^-_{t_k} - \mathbf{K}_{t_k}\left(\mathbf{H}\mathbf{P}^-_{t_k}\mathbf{H} + \mathbf{R}\right)^{-1}\mathbf{K}_{t_k} \\ &\approx \mathbf{P}^-_{t_k} - \mathbf{K}_{t_k}\left(\mathbf{H}\mathbf{P}^-_{t_k}\mathbf{H}\right)\mathbf{K}_{t_k} \\ &\approx 0.\end{aligned} \tag{3.19}$$

so that the posterior variance (i.e. the uncertainty about the position of $\mathbf{X}_{t_k}$) is low.

A similar analysis can be carried out in several dimensions. However, it is complicated by the fact that one must first choose a suitable matrix norm to be able to quantify a statement like '$\mathbf{R}$ is small'. Furthermore, some components of the state may not be observable (i.e. $\mathbf{H}$ is low-rank). When this is the case, one must rely on knowledge of the correlation structure of $\mathbf{X}$ to deduce information about its hidden states.

We will now relate the preceding discussion to the so-called 'continuous-discrete filtering problem', in which a continuous process is observed at discrete time intervals. The so-called Ornstein-Uhlenbeck SDE satisfies

$$d\mathbf{X}_t = \mathbf{a}(\theta - \mathbf{X}_t)dt + \mathbf{b}d\mathbf{W}_t. \tag{3.20}$$

The Ornstein-Uhlenbeck process is the stochastic counterpart of a standard linear ODE. Here, $\mathbf{X}_t \in \mathbb{R}^n$. When the matrix $\mathbf{a} \in \mathbf{R}^{n\times n}$ is negative-definite, the parameter $\theta \in \mathbb{R}^n$ plays the role of a 'long-run mean' about which the process oscillates.

The Solution of the Ornstein-Uhlenbeck process is

$$\mathbf{X}_{t_k} = e^{-\mathbf{a}(t_k - t_{k-1})}\mathbf{X}_{t_{k-1}} + \int_{t_{k-1}}^{t_k} e^{\mathbf{a}(u-t_k)}\mathbf{a}\theta du + \int_{t_{k-1}}^{t_k} e^{\mathbf{a}(u-t_k)}\mathbf{b}d\mathbf{W}_u. \tag{3.21}$$

When $\mathbf{X}$ is multidimensional, the exponentials on the left-hand side should be interpreted as matrix exponentials. The solution can be written in the form of equation (3.1) by setting

$$\mathbf{A} = e^{-\mathbf{a}(t_k - t_{k-1})}, \tag{3.22}$$

$$\mathbf{c} = \int_{t_{k-1}}^{t_k} e^{\mathbf{a}(u-t_k)}\mathbf{a}\theta du, \tag{3.23}$$

$$\mathbf{Z}_{t_{k-1}} = \int_{t_{k-1}}^{t_k} e^{\mathbf{a}(u-t_k)}\mathbf{b}d\mathbf{W}_u. \tag{3.24}$$

Thus, the continuous-discrete filtering problem for an Ornstein-Uhlenbeck process can be formulated in terms of the standard discrete-time Kalman filter, provided that one is only interested in the state of the system at observation times $\{t_i\}$.

3.1.1 The Extended Kalman filter

The preceding discussion focused on *linear* models. However, many systems of interest do not have linear dynamics. We now discuss how it is possible to salvage some of the insight developed from the linear Kalman filter. Under certain circumstances, non-linear dynamical systems can be approximated with linear systems without inducing

much error. Indeed, the tracking and dynamics equations used in the Apollo program were nonlinear, whereas a linear approximation to those equations was implemented in practice.

To build intuition for when it might be appropriate to approximate a nonlinear SDE with a linear SDE, we consider the generic system

$$d\mathbf{X}_t = \mathbf{a}(\mathbf{X}_t)dt + \mathbf{b}(\mathbf{X}_t)d\mathbf{W}_t. \tag{3.25}$$

In the general case, $\mathbf{X}$ is clearly nonlinear and non-Gaussian. In order for the Kalman filter to be applicable, we can attempt to construct a process $\tilde{\mathbf{X}}$ that captures some of the properties of $\mathbf{X}$. The dynamics of the linear approximation take the form

$$d\tilde{\mathbf{X}}_t = \mathbf{A}\tilde{\mathbf{X}}_t dt + \mathbf{c}dt + \mathbf{B}d\mathbf{W}_t \tag{3.26}$$

Perhaps the simplest way of choosing $\mathbf{A}$, $\mathbf{c}$, and $\mathbf{B}$ is to hold $\mathbf{b}$ constant at some point $\mathbf{p}$, and to Taylor expand $\mathbf{a}(\cdot)$ about that point. That is,

$$\mathbf{a}(\mathbf{p}+\mathbf{h}) \approx \mathbf{a}(\mathbf{p}) + \mathbf{J}_{\mathbf{p}}\mathbf{h}, \tag{3.27}$$

where $\mathbf{J}$ is the Jacobian matrix of $\mathbf{a}$ at $\mathbf{p}$.

One then has

$$\mathbf{A} = \mathbf{J}_{\mathbf{p}}, \tag{3.28}$$

$$\mathbf{c} = \mathbf{a}(\mathbf{p}), \tag{3.29}$$

$$\mathbf{B} = \mathbf{b}(\mathbf{p}), \tag{3.30}$$

so that

$$d\tilde{\mathbf{X}}_t = \mathbf{J}_{\mathbf{p}}\tilde{\mathbf{X}}_t dt + \mathbf{a}(\mathbf{p})dt + \mathbf{b}(\mathbf{p})d\mathbf{W}_t. \tag{3.31}$$

When the Kalman filter is applied to a nonlinear stochastic process in conjunction with the linear approximation (3.31), the algorithm is known as the *extended Kalman filter*.

One natural question to ask is 'under what circumstances is this a good approximation?'. It is clear that the error is induced by holding $\mathbf{b}(\cdot)$ constant and Taylor expanding $\mathbf{a}(\cdot)$. If $\mathbf{b}$ is 'almost constant' in the sense that its variation is low, and $\mathbf{a}$ is 'almost linear' in the sense that the variation of its Jacobian is low, then $\mathbf{X}$ behaves 'almost' like a linear process. It is rather complicated to make precise what is meant by 'almost' here, but see [29, Ch. 8.1] for a rigorous example.

The other possibility is that the process $\mathbf{X}$ is constrained to stay in an area where the approximation (3.27) is valid. One very common way of enforcing this constraint

is to ensure that the observations are precise and close together. If the observations are precise, then the filtering distribution will be highly peaked around its mean, and we can be confident that the signal is nearby. For this reason, we take the Taylor expansion about the mean of the filtering distribution at time T_{k-1}. By continuity of the process $\mathbf{X}$, it follows that $\mathbf{X}_{t_k}$ is close to $\mathbf{X}_{t_{k-1}}$ when $t_k - t_{k-1}$ is small. Thus the process does not have time to diffuse into a region where the approximation is poor before the next observation is made.

If neither of these conditions hold, so that $\mathbf{X}$ is highly nonlinear and the observations are noisy or spaced far apart in time, the extended Kalman filter can fail catastrophically. For this reason, many improvements on the EKF have been proposed in the literature. We now review one such improvement: the *unscented Kalman filter* (UKF). To this end, we first discuss a method for constructing Gaussian approximations to certain non-Gaussian random variables. The method is known as the *unscented transform*.

3.1.2 The unscented transform

Given a Gaussian random variable $\mathbf{Z}$ and a function f, the random variable $f(\mathbf{Z})$ is non-Gaussian in general. In order to find a Gaussian approximation to $f(\mathbf{Z})$, one has two options: modify some property of f, or modify some property of $\mathbf{Z}$. The extended Kalman filter relies on the first of these options: $f(x)$ is replaced with $f(a) + (x - a)f'(a)$. Another possibility is to approximate $\mathbf{Z}$ with a number of point masses that capture its distribution. For example, one might use the Monte-Carlo approximation

$$p_{\mathbf{Z}}(x) \approx \frac{1}{N}\sum_{i=1}^{N} \delta_{\mathbf{Z}^i}(x), \tag{3.32}$$

$$\mathbb{E}[f(\mathbf{Z})] \approx \frac{1}{N}\sum_{i=1}^{N} f(\mathbf{Z}^i), \tag{3.33}$$

$$\mathrm{Var}(f(\mathbf{Z})) \approx \frac{1}{N}\sum_{i=1}^{N} f(\mathbf{Z}^i)f(\mathbf{Z}^i)^\top - \left(\frac{1}{N}\sum_{i=1}^{N} f(\mathbf{Z}^i)\right)\left(\frac{1}{N}\sum_{i=1}^{N} f(\mathbf{Z}^i)\right)^\top, \tag{3.34}$$

where $\{\mathbf{Z}^i\}$ are i.i.d samples from $\mathbf{Z}$. The critical insight here is that, in some sense, it is *easier to approximate* $\mathbf{Z}$ *than it is to approximate* f.

In time-critical applications, or in situations where limited computational power is available, the Monte-Carlo approach may be prohibitively expensive. This is especially true if f is expensive to evaluate. One can lower N, the number of samples, but this

increases the variance of the estimates (3.33) and (3.34). One solution, put forward by S.J Julier in his Ph.D thesis [30] is to use a weighted collection of deterministically-chosen points to approximate the distribution of $\mathbf{Z}$. These so-called 'sigma points' are chosen to capture as faithfully as possible the statistics of $\mathbf{Z}$ while minimising the computational cost of doing so. In the so-called *unscented transform*, one uses $2n+1$ points, where n is the dimension of $\mathbf{Z}$.

We will restrict our exposition to the case where $\mathbf{Z}$ has an n-dimensional multivariate normal distribution, and we wish to fit a multivariate normal distribution to $f(\mathbf{Z})$. Suppose $\mathbf{Z}$ has mean $\mathbf{m}$ and covariance $\mathbf{P}$. The sigma points for the unscented transform are constructed as follows. One chooses two tuning parameters α and κ, then sets $\lambda = \alpha^2(n+\kappa) - n$. The sigma points are then defined by the following expressions:

$$\sigma^0 = \mathbf{m}, \tag{3.35}$$

$$\sigma^i = \mathbf{m} + (\sqrt{(n+\lambda)\mathbf{P}})_{*i}, \quad 1 \le i \le n, \tag{3.36}$$

$$\sigma^{n+i} = \mathbf{m} - (\sqrt{(n+\lambda)\mathbf{P}})_{*i}, \quad 1 \le i \le n. \tag{3.37}$$

Here $(\sqrt{\mathbf{P}})_{*i}$ is the i-th column of a choice of the matrix square root of $\mathbf{P}$ defined via $\mathbf{P} = \sqrt{\mathbf{P}}\sqrt{\mathbf{P}}^\top$. The sigma points are determined once one chooses an appropriate matrix square root.

The mean and covariance of $f(\mathbf{Z})$ are approximated by a weighted average of the sigma-point images. Define $\mathcal{Y}_i = f(\sigma^i)$, and set

$$\mathbb{E}[f(\mathbf{Z})] \simeq \mu = \sum_{i=0}^{2n} w_i^{(m)} \mathcal{Y}_i. \tag{3.38}$$

We can then make the approximations

$$\mathrm{Cov}[f(\mathbf{Z})] \simeq \mathbf{S} = \sum_{i=0}^{2n} w_i^{(c)} \left(\mathcal{Y}_i - \mu\right)\left(\mathcal{Y}_i - \mu\right)^\top, \tag{3.39}$$

$$\mathrm{Cov}[\mathbf{Z}, f(\mathbf{Z})] \simeq \mathbf{C} = \sum_{i=0}^{2n} w_i^{(c)} \left(\sigma^i - \mathbf{m}\right)\left(\mathcal{Y}_i - \mu\right)^\top. \tag{3.40}$$

The weights depend on a third tuning parameter β, and are given by

$$\begin{aligned} w_0^{(m)} &= \frac{\lambda}{n+\lambda}, \\ w_0^{(c)} &= \frac{\lambda}{n+\lambda} + (1-\alpha^2+\beta), \\ w_i^{(m)} &= \frac{1}{2(n+\lambda)} \qquad i = 1,\ldots,2n, \\ w_i^{(c)} &= \frac{1}{2(n+\lambda)} \qquad i = 1,\ldots,2n. \end{aligned} \tag{3.41}$$

It is well known that the unscented transform matches the mean of $f(\mathbf{Z})$ exactly when f is a polynomial of degree three or less. In general, errors in the estimate of the mean are introduced only by the fourth and higher terms in the Taylor expansion of f [31].

3.2 Sigma point Kalman filters for diffusion processes

The unscented Kalman filter (UKF) and the extended Kalman filter are both examples of *Gaussian filters*, where the filtering distribution at time t is approximated by a Gaussian distribution. The filtering problem is thus reduced to approximation of the conditional mean and covariance of the filtering distribution:

$$\mathbf{m}_t = \mathbb{E}\left[\mathbf{X}_t \mid \{\mathbf{Y}_{t_k} : t_k \leq t\}\right] \tag{3.42}$$

and

$$\mathbf{P}_t = \mathrm{Cov}\left[\mathbf{X}_t \mid \{\mathbf{Y}_{t_k} : t_k \leq t\}\right]. \tag{3.43}$$

It is usually necessary to approximate the conditional mean and covariance: for a general nonlinear diffusion, the moments are only known in terms of the solution of the Fokker-Planck equation, described in Section 2.2.4 [32, 10]. In dimensions higher than three, the Fokker-Planck equation is typically numerically intractable.

The simplest application of the UKF to a diffusion relies on discretisation of the process. Suppose that at time t_{k-1} we have an estimate of $\mathbf{m}_{t_{k-1}}$ and $\mathbf{P}_{t_{k-1}}$. In the prediction step, our aim is to compute an estimate of $\mathbf{m}_t$ and $\mathbf{P}_t$ at time $t = t_k^-$, the instant before the next observation arrives.

We divide the time interval $[t_{k-1}, t_k]$ into a number of sub-intervals of length Δt (for clarity, we will discuss the interval $[0, t_1]$ here). We then approximate the SDE (3.25) on the grid $\{\mathbf{X}_{\Delta t}, \mathbf{X}_{2\Delta t}, \dots\}$ via the relation

$$\mathbf{X}_{(j+1)\Delta t} = \mathbf{f}(\mathbf{X}_{j\Delta t}, \mathbf{Z}_j), \tag{3.44}$$

where $\mathbf{Z}_0, \mathbf{Z}_1, \dots$ is a suitable sequence of Gaussian random variables. Here, $\mathbf{f}$ is a transition function that depends on the method of discretisation, and $\mathbf{Z}_k$ is typically draw from a spherical Gaussian distribution of dimension d. For example, in the *Euler-Maruyama* scheme [22],

$$\mathbf{f}(\mathbf{X}_{j\Delta t}, \mathbf{Z}_j) = \mathbf{X}_{j\Delta t} + \mathbf{a}(\mathbf{X}_{j\Delta t})\Delta t + \mathbf{b}(\mathbf{X}_{j\Delta t})\sqrt{\Delta t}\mathbf{Z}_j, \tag{3.45}$$

where $\mathbf{Z}_j \sim \mathcal{N}(0, \mathbf{I}_d)$.

In this sense, $\mathbf{X}_{(j+1)\Delta t}$ is the image of $(\mathbf{X}_{j\Delta t}, \mathbf{Z}_j)$ under a nonlinear transform $\mathbf{f}$. Given a Gaussian approximation to $\mathbf{X}_{j\Delta t}$, one can apply the unscented transform to $\mathbf{f}$ to find a Gaussian approximation of $\mathbf{X}_{(j+1)\Delta t}$. One proceeds iteratively until t_k, at which point the prediction phase ends and we proceed to the update phase. Instead of the Euler–Maruyama method, one can in some circumstances use higher order Itô–Taylor expansions, stochastic Runge–Kutta methods or various other methods [22].

Alternatively, one can take a limit as $\Delta t \to 0$ instead of iteratively applying the unscented transform at the prediction. By doing so, one recovers a system of differential equations for the predictive mean and covariance (see, e.g., [33, 34]):

$$\begin{aligned}
\frac{d\mathbf{m}_t^-}{dt} &= \mathbb{E}[\mathbf{a}(\mathbf{X}_t^-)] \\
\frac{d\mathbf{P}_t^-}{dt} &= \mathbb{E}[\mathbf{a}(\mathbf{X}_t^-)(\mathbf{X}_t^- - \mathbf{m}_t^-)^\top] \\
&\quad + \mathbb{E}[(\mathbf{X}_t^- - \mathbf{m}_t^-)\mathbf{a}^\top(\mathbf{X}_t^-)] \\
&\quad + \mathbb{E}[\mathbf{b}(\mathbf{X}_t^-)\mathbf{b}^\top(\mathbf{X}_t^-)],
\end{aligned} \tag{3.46}$$

where the expectations are taken with respect to the Gaussian distribution $\mathbf{X}_t^- \sim \mathcal{N}(\mathbf{m}_t^-, \mathbf{P}_t^-)$. In chapter 7, we propose a novel method for computing the predictive distribution.

When a new observation is made (at time $t = t_k$, say), we must update our predictive distribution with the new information that has arrived. Let $\mathbf{m}_{t_k}^-$ and $\mathbf{P}_{t_k}^-$ be the mean and covariance of the predictive distribution immediately before the new observation arrives. We form an approximation $\hat{\mathbf{X}}_{t_k}^-$ of the signal, which is Gaussian with the predictive mean and covariance. The update equations are similar to those of the linear Kalman filter, but are included here for completeness:

$$\begin{aligned}
\mu_k &= \mathbb{E}[h(\hat{\mathbf{X}}_{t_k}^-)] \\
\mathbf{S}_k &= \mathbb{E}[(h(\hat{\mathbf{X}}_{t_k}^-) - \mu_k)(h(\hat{\mathbf{X}}_{t_k}^-) - \mu_k)^\top] + \mathbf{R}_k \\
\mathbf{C}_k &= \mathbb{E}[(\hat{\mathbf{X}}_{t_k}^- - \mathbf{m}_{t_k}^-)(h(\hat{\mathbf{X}}_{t_k}^-) - \mu_k)^\top] \\
\mathbf{K}_k &= \mathbf{C}_k \mathbf{S}_k^{-1} \\
\mathbf{m}_{t_k} &= \mathbf{m}_{t_k}^- + \mathbf{K}_k(\mathbf{Y}_{t_k} - \mu_k) \\
\mathbf{P}_{t_k} &= \mathbf{P}_{t_k}^- - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^\top,
\end{aligned} \tag{3.47}$$

The updated distribution has mean $\mathbf{m}_{t_k}$ and covariance $\mathbf{P}_{t_k}$. When the observation function $\mathbf{h}$ is nonlinear, one can apply the unscented transform to $\mathbf{h}(\hat{\mathbf{X}}_{t_k}^-)$ to compute an approximation of μ_k, $\mathbf{S}_k$, and $\mathbf{C}_k$ in (3.47). More complex update rules that have been tuned for numerical stability are also known in the literature [26]. These are also applicable to the linear Kalman filter.

3.3 Particle filters

Sequential Monte-Carlo (SMC) methods are a powerful and general family of techniques for generating samples from high-dimensional probability distributions. Among the best-known applications of SMC is the *particle filter*. Particle filters offer a promising means of approximating the optimal solution of the filtering problem. A number of excellent tutorials and reviews are avaiable: see, for example, [35] [36] [37], and [38] for a textbook-length collection of important developments up to 2001. In this section, we discuss the use of the particle filter when the signal of interest is a nonlinear multivariate diffusion process.

Roughly speaking, the particle filter works as follows. At time t_{k-1}, we assume we have a collection of weighted point masses, or 'particles' that approximate the filtering distribution. The particles are propagated forward in time using a suitable transition density, resulting in a new collection of point masses. These are assigned a new set of weights by means of importance sampling. The weights are computed using knowledge of the signal dynamics, the observation at time t_k and the importance transition density.

The *bootstrap filter* [39] is discussed in detail in Section 3.3.1. It is one simple implementation of a particle filter. In the bootstrap filter, one uses the prior dynamics of the signal to propagate the particles forward in time. The bootstrap filter can fail when the conditional distribution $p(\mathbf{X}_t|\mathbf{Y}_t, \mathbf{X}_{t_{k-1}})$ is tightly constrained relative to the unconditional distribution $p(\mathbf{X}_{t_k}|\mathbf{X}_{t_{k-1}})$. This is because most of the particles are assigned a low weight in the importance sampling step. In the worst case, one approximates the filtering distribution with what amounts to a single particle. This phenomenon tends to occur in high-dimensional filtering problems, and when the observations are highly informative about the state of the system [40].

One way of correcting for the degeneracy issue is to employ better importance distributions. For example, one can construct a Gaussian approximation to the joint distribution of the signal and the observations at time t_k. One can then use standard results about the multivariate normal distribution to approximate the distribution of the signal conditioned on the observation. The conditioned distribution is often an effective importance distribution for the particle filter.

The Gaussian approximation can be constructed by linearising the signal and observation functions. Over short timescales, or for processes whose dynamics are 'almost' linear, this can be an effective strategy. However, this is not always the case, and per-

formance can be poor. An alternative is to use the *unscented transform* [41] [42] [31], which often produces a more accurate approximation. This is known as the *Unscented particle filter* [43].

3.3.1 The bootstrap filter

In this section, we will briefly review the ideas behind sequential importance sampling, and particle filtering in particular. We will assume that our signal, $\mathbf{X}$, is a diffusion process satisfying the stochastic differential equation

$$d\mathbf{X}_t = \mathbf{a}(\mathbf{X}_t)dt + \mathbf{b}(\mathbf{X}_t)d\mathbf{W}_t, \qquad \mathbf{X}_0 \sim p_{X_0}. \tag{3.48}$$

Here, $\mathbf{X}_t \in \mathbb{R}^n$, $\mathbf{W}_t \in \mathbb{R}^d$, $\mathbf{a} : \mathbb{R}^n \to \mathbb{R}^n$, and $\mathbf{b} : \mathbb{R}^n \to \mathbb{R}^{n\times d}$. We assume $\mathbf{a}$ and $\mathbf{b}$ satisfy the usual conditions that ensure $\mathbf{X}$ has a unique solution (for example, the assumption that they are globally Lipschitz is strong enough). As usual, we assume that noisy observations of the process $\mathbf{X}$ are made at times $t_{1:k}$.

Observations are assumed to arrive in the form

$$\mathbf{Y}_{t_k} = \mathbf{h}(\mathbf{X}_{t_k}) + \varepsilon_{t_k}, \tag{3.49}$$

where $\mathbf{h} : \mathbb{R}^n \to \mathbb{R}^m$, and $\mathbf{Y}_t \in \mathbb{R}^m$. The random variables $\{\varepsilon_{t_i}\}$ are jointly Gaussian and independent, each with mean 0 and covariance $\mathbf{R}$. We assume a fixed time $T = t_k - t_{k-1}$ between observations for clarity of exposition, though this is easy to generalise.

Diffusion processes possess the Markov property, so that the joint distribution of state and observations factorises as

$$p(\mathbf{X}_{t_{1:k}}, \mathbf{Y}_{t_{1:k}}|\mathbf{X}_0) = \prod_{i=1}^{k} p(\mathbf{Y}_{t_i}|\mathbf{X}_{t_i})p(\mathbf{X}_{t_i}|\mathbf{X}_{t_{i-1}}). \tag{3.50}$$

In order to compute the transition density $p(\mathbf{X}_{t_k}|\mathbf{X}_{t_{k-1}})$ for a general nonlinear diffusion, one must solve a partial differential equation known as the *Fokker-Planck equation*. It is only in special cases that this solution is available in closed form, and one must typically resort to approximations of some sort.

The update step for the filter can be described in terms of the transition density of $\mathbf{X}$ via the integral equation

$$p(\mathbf{X}_{t_k}|\mathbf{Y}_{t_{1:k-1}}) = \int p(\mathbf{X}_{t_k}|\mathbf{X}_{t_{k-1}})p(\mathbf{X}_{t_{k-1}}|\mathbf{Y}_{t_{1:k-1}})d\mathbf{X}_{t_{k-1}}. \tag{3.51}$$

This is essentially the *Chapman-Kolmogorov equation* from the theory of Markov processes.

The observation at time t_k can be incorporated into the posterior distribution via an application of Bayes' theorem as follows:

$$\begin{aligned} p(\mathbf{X}_{t_k}|\mathbf{Y}_{t_{1:k-1}}, \mathbf{Y}_{t_k}) &= \frac{p(\mathbf{X}_{t_k}, \mathbf{Y}_{t_k}|\mathbf{Y}_{t_{1:k-1}})}{p(\mathbf{Y}_{t_k}|\mathbf{Y}_{t_{1:k-1}})} \\ &= \frac{p(\mathbf{Y}_{t_k}|\mathbf{X}_{t_k})p(\mathbf{X}_{t_k}|\mathbf{Y}_{t_{1:k-1}})}{p(\mathbf{Y}_{t_k}|\mathbf{Y}_{t_{1:k-1}})}. \end{aligned} \tag{3.52}$$

The prediction step relies on the intractable transition density $p(\mathbf{X}_{t_k}|\mathbf{X}_{t_{k-1}})$. In addition, the normalising constant $p(\mathbf{Y}_{t_k}|\mathbf{Y}_{t_{1:k-1}})$ in the update step is typically difficult to compute when the state dimension of $\mathbf{X}$ is large. In the next section, we will see how importance sampling can be exploited to overcome both of these issues.

3.3.2 Sequential importance sampling and the bootstrap filter

As we noted earlier, it is usually impractical to evaluate the transition density of a general diffusion process at a given time. However, it is relatively straightforward to construct an approximate sample path of a diffusion process. This can be achieved by means of a numerical scheme such as the Euler-Maruyama method or the Ito-Taylor scheme. In some cases is is even possible to use a form of rejection sampling to generate sample paths from a nonlinear diffusion without inducing any discretisation bias [44] [45].

We will now show how sequential importance sampling (SIS) can be applied to a diffusion process in order to compute an approximate solution to the filtering problem. Monte-Carlo methods such as SIS are particularly suited to filtering problems involving diffusion processes. This is due to the relative ease of generating approximate sample paths, and the presence of the Markov property.

Suppose at time t_{k-1}, we have a collection of 'particles' $\{\mathbf{X}^i_{t_{k-1}}\}$ and positive weights $\{w^i_{t_{k-1}}\}$ that sum to unity, such that the filtering distribution $p(\mathbf{X}_{t_{k-1}}|\mathbf{Y}_{t_{1:k-1}})$ is approximated by

$$p(\mathbf{X}_{t_{k-1}}|\mathbf{Y}_{t_{1:k-1}}) \approx \sum_i w^i_{t_{k-1}} \delta_{\mathbf{X}^i_{t_{k-1}}}. \tag{3.53}$$

A weighted sample such as this can be initialised at time 0 by drawing uniformly-weighted samples from the prior distribution on $\mathbf{X}_0$.

We cannot apply the prediction equation in a straightforward manner, since the transition density is not generally available in closed form. However, it is usually straightforward to generate samples from $p(\mathbf{X}_{t_k}|\mathbf{X}_{t_{k-1}})$ (provided that one is prepared to tolerate some discretisation bias). In the simplest version of the bootstrap filter,

we 'move' each particle $\delta_{\mathbf{X}^i_{t_{k-1}}}$ in turn by drawing a sample $\mathbf{X}^i_{t_k}$ from the distribution $p(\mathbf{X}_{t_k}|\mathbf{X}^i_{t_{k-1}})$. This gives a weighted particle approximation for the predictive distribution:

$$p(\mathbf{X}_{t_k}|\mathbf{Y}_{t_{1:k-1}}) \approx \sum_i w^i_{t_{k-1}} \delta_{\mathbf{X}^i_{t_k}}. \tag{3.54}$$

At time t_k, the predictive distribution must be updated with the observation $\mathbf{Y}_{t_k}$. We perform this update by re-weighting the particles in (3.54). Equation (3.52) shows us that, up to normalisation, the weights should satisfy

$$\tilde{w}^i_{t_k} = w^i_{t_{k-1}} p(\mathbf{Y}_{t_k}|\mathbf{X}^i_{t_{k-1}}) \tag{3.55}$$

The weights are now normalised by setting $w^i_{t_k} = \tilde{w}^i_{t_k} / \sum \tilde{w}^j_{t_k}$, and we arrive at the new weighted particle approximation

$$p(\mathbf{X}_{t_k}|\mathbf{Y}_{t_{1:k}}) \approx \sum w^i_{t_k} \delta_{\mathbf{X}^i_{t_k}}. \tag{3.56}$$

Repeated application of the weight update equation (3.55) as described above can be problematic. Typically, one of the weights will grow to dominate the others, with the result that the filtering distribution is effectively approximated by a single particle. This issue can be resolved by adding a *resampling* step. When the weights become sufficiently imbalanced, we create a new, uniformly weighted collection of particles by sampling from the discrete distribution in (3.53) with replacement. The prediction and update steps are then applied as usual. This is known as the *bootstrap filter*.

Instead of sampling from the prior dynamics of the particles to generate the approximation to the predictive equation in (3.54), one can in principal draw from an alternative distribution $q(\mathbf{X}_{t_k}|\mathbf{X}_{t_{k-1}}, \mathbf{Y}_{t_k})$ and apply importance sampling. The predictive approximation is then given by

$$p(\mathbf{X}_{t_k}|\mathbf{Y}_{t_{1:k-1}}) \approx \sum_i w^i_{t_{k-1}} \frac{p(\mathbf{X}^i_{t_k}|\mathbf{X}^i_{t_{k-1}})}{q(\mathbf{X}^i_{t_k}|\mathbf{X}^i_{t_{k-1}}, \mathbf{Y}_{t_k})} \delta_{\mathbf{X}^i_{t_k}}. \tag{3.57}$$

To update the predictive equation, we apply the relation

$$\tilde{w}^i_{t_k} = w^i_{t_{k-1}} p(\mathbf{Y}_{t_k}|\mathbf{X}^i_{t_{k-1}}) \frac{p(\mathbf{X}^i_{t_k}|\mathbf{X}^i_{t_{k-1}})}{q(\mathbf{X}^i_{t_k}|\mathbf{X}^i_{t_{k-1}}, \mathbf{Y}_{t_k})}, \tag{3.58}$$

and normalise the weights as usual.

The possibility of using importance sampling comes with the caveat that the *ratio* of densities must be tractable, even if the transition density itself is not. This is a severe constraint, and few proposals are suitable. We explore one possible proposal distribution in Chapter 8.

3.3.3 The unscented particle filter

Doucet, et al. [46] have shown that the optimal importance distribution in (3.57) is

$$q(\mathbf{X}^i_{t_k}|\mathbf{X}^i_{t_{k-1}}, \mathbf{Y}_{t_k}) = p(\mathbf{X}^i_{t_k}|\mathbf{X}^i_{t_{k-1}}, \mathbf{Y}_{t_k}). \tag{3.59}$$

That is, the optimal importance distribution at time t_k is the filtering distribution conditioned on the incoming observation at time t_k. The distribution is optimal in the sense that it minimises variance of the importance weights. It also has the property that the importance weights at time t_k depend only on the state of the particle at time t_{k-1}.

It is often infeasible to sample from this distribution, though one can attempt to sample from a distribution that approximates it. One method of doing so uses the *unscented transform* to approximate the mean and variance of the optimal importance distribution. A Gaussian distribution with the corresponding mean and variance is used in place of the optimal distribution. This is known as the *unscented particle filter* [47].

Chapter 4

Parameter estimation

In this chapter, we review and summarise a number of works related to parameter estimation of diffusion processes. Some methods aim to compute the full Bayesian posterior distribution of the data, whereas others aim to maximise the likelihood function. In the first section, we discuss Bayesian methods, while maximum likelihood approximations are discussed in Section 4.2. We point the reader to the review paper of Sorensen [48], who discusses parameter estimation in the univariate stationary setting. Other useful review articles include Singer [49], and the chapter on estimation in Iacus [50]. Important early work on this topic was undertaken by Florens-Zmirou [51] [52].

4.1 Bayesian inference

As we mentioned in Section 1.2, one is often interested in estimating some parameter vector θ that governs the behaviour of a diffusion process $\mathbf{X}$. In when the observations of $\mathbf{X}$ are noisy, one has

$$\begin{aligned} p(\theta|\mathbf{Y}_{t_{1:n}}) &\propto p(\theta)p(\mathbf{Y}_{t_{1:n}}|\theta) && (4.1)\\ &= p(\theta)\int p(\mathbf{Y}_{t_{1:n}},\mathbf{X}_{t_{1:n}}|\theta)d\mathbf{X}_{t_{1:n}} \\ &= p(\theta)\int \prod_{k=1}^{n} p(\mathbf{Y}_{t_k}|\mathbf{X}_{t_k})p(\mathbf{X}_{t_k}|\mathbf{X}_{t_{k-1}},\theta)d\mathbf{X}_{t_{1:n}}. && (4.2)\end{aligned}$$

When the observations are exact, the posterior distribution has a particularly simple

form:

$$\begin{aligned} p(\theta|\mathbf{X}_{t_{1:n}}) &\propto p(\theta)p(\mathbf{X}_{t_{1:n}}|\theta) \\ &= p(\theta)\prod_{k=1}^{n} p(\mathbf{X}_{t_k}|\mathbf{X}_{t_{k-1}},\theta). \end{aligned} \tag{4.3}$$

This special case is useful for outlining the problems that one can encounter when attempting to infer the posterior distribution over θ.

It is somewhat counter-intuitive, but the addition of observation noise can often make it *easier* to sample from the posterior distribution. The reason for this is that when observations are noisy, one can use standard numerical methods to generate data samples that are consistent with the observations. Any sample paths that lie in the rough neighbourhood of the observations are plausible. On the other hand, when observations are precise, most sample paths have low likelihood. Thus, unconditioned sample paths are not suitable for use in inference algorithms in this setting.

For certain diffusions such as the Ornstein-Uhlenbeck process, the transition density $p(\mathbf{X}_{t_k}|\mathbf{X}_{t_{k-1}})$ can be computed in closed form. Hence it is possible to evaluate the posterior distribution, at least up to some normalising constant. For example, Stimberg et al. [53] formulate a model which follows an Ornstein-Uhlenbeck process, the parameters of which vary in time and are governed by a latent Markov jump process. Since the model is linear and Gaussian when conditioned on the value of this Markov jump process, inference can be conducted exactly.

For a general nonlinear diffusion, this is not the case. The transition density cannot be evaluated without significant computational expense, rendering (4.2) and (4.3) intractable. However, nonlinear diffusions do possess useful structure than can be exploited. The Euler-Maruyama discretisation shows us that, in some sense, diffusion processes are *locally* Gaussian. Given a diffusion with drift $\mathbf{a}_\theta(x)$ and diffusion coefficient $\mathbf{b}_\theta(x)$, the Euler-Maruyama approximation tells us that when $\Delta t = t_k - t_{k-1}$ is small,

$$p(\mathbf{X}_{t_k}|\mathbf{X}_{t_{k-1}}) \approx \mathcal{N}\Big(\mathbf{X}_{t_k}|\mathbf{X}_{t_{k-1}} + \mathbf{a}_\theta(\mathbf{X}_{t_{k-1}})\Delta t, \mathbf{b}_\theta(\mathbf{X}_{t_{k-1}})\mathbf{b}_\theta(\mathbf{X}_{t_{k-1}})^\top \Delta t\Big). \tag{4.4}$$

For this reason, it is sensible to augment the data, say $(\mathbf{X}_{t_{k-1}}, \mathbf{X}_{t_k})$, with 'fictional' intermediate data points. We suppose there are M additional data points observed times $t_{k-1} < t_{k-1,1} < \cdots < t_{k-1,M} < t_k$ which are sufficiently close to one another that (4.4) is a good approximation. The additional data can be integrated out by Monte-Carlo methods. In summary, between any two observations we work with an augmented data

set $(\mathbf{X}_{t_{k-1}}, \mathbf{X}_{t_{k-1,1}}, \dots, \mathbf{X}_{t_{k-1,M}}, \mathbf{X}_{t_k})$. In the augmented space, each data point is approximately distributed according to a Gaussian distribution conditional on the previous one, but the augmentation also raises the dimensionality of the inference problem, which can be undesirable. Perhaps the most common strategy in the literature is to integrate out the augmented data using Markov chain Monte Carlo methods. However, Fearnhead [54] discusses a number of other methods (such as sequential Monte Carlo and importance sampling) that are also applicable.

The data augmentation strategy forms the basis of a number of papers on MCMC and parameter estimation, which we will now review. Elerian, Chib, and Shepard [55] consider the case of a univariate process $\mathbf{X}$ observed without error. We review this first paper in some depth since it demonstrates the main ideas in the MCMC inference paradigm and the problems encountered therein, without adding extra complications (such as multivariate, partially observed data). Similar methodology was also proposed by Eraker [56] with application to financial modelling.

We will use the notation $\mathbf{X}^*_{t_{k-1}} = (\mathbf{X}_{t_{k-1,1}}, \dots, \mathbf{X}_{t_{k-1,M}})$ to refer to the imputed data between the observations at time t_{k-1} and t_k. The combination of imputed and observed data, $(\mathbf{X}_{t_1}, \mathbf{X}^*_{t_1}, \mathbf{X}_{t_2}, \mathbf{X}^*_{t_2}, \dots)$ will be referred to as the augmented data.

Elerian, Chib, and Shepard implement a Gibbs sampler that alternates between updating θ conditional on the augmented data, and updating the imputed data conditional on θ and the observed data. As a result of the Markov property of $\mathbf{X}$, $\mathbf{X}^*_{t_i}$ is independent from $\mathbf{X}^*_{t_j}$ conditional on any observation between times t_i and t_j. That is,

$$p(\mathbf{X}^*_{t_1}, \dots, \mathbf{X}^*_{t_n} | \mathbf{X}_{t_{1:n}}, \theta) = \prod_{i=1}^{n} p(\mathbf{X}^*_{t_i} | \mathbf{X}_{t_i}, \mathbf{X}_{t_{i+1}}, \theta). \tag{4.5}$$

Hence, when updating the augmented data, it is sufficient to be able to sample from $p(\mathbf{X}^*_{t_i} | \mathbf{X}_{t_i}, \mathbf{X}_{t_{i+1}}, \theta)$ for each i sequentially.

While it is easy to sample from a diffusion process given its initial condition, it is highly nontrivial to sample from a diffusion given an initial *and final* condition, as we saw in Section 2.2.6. A diffusion whose initial and final conditions are specified is known as a *diffusion bridge*. One can argue that the ability to generate samples cheaply from a general diffusion bridge would render most problems in filtering, smoothing, and parameter estimation of diffusion processes relatively trivial. A diffusion bridge can itself be described as a stochastic differential equation. Its dynamics coincide with the dynamics of the unconditioned diffusion, but with a new 'potential well' term appearing in the drift of the process (see Section 2.2.6). The new term is given in terms of the transition density of the unconditioned process, which is intractable and must

typically be approximated.

Elerian, Chib and Shepard advocate using Newton-Raphson iteration to find a local maximum of $\log(p(\mathbf{X}^*_{t_i}|\mathbf{X}_{t_i},\mathbf{X}_{t_{i+1}},\theta))$ (and hence a mode of the distribution). The iteration is also used to compute the Hessian of the distribution at the local minimum, which gives an estimate of the distribution's covariance. These data are used to construct a multivariate Gaussian approximation to the distribution. The approximation is used as a proposal distribution in a Metropolis-Hastings correction step.

When M is large (i.e. a large number of latent variables are used), the Gaussian approximation may not be suitable, and the rate of rejection can rise to unacceptable levels. In the numerical example considered in the paper, performance declines between $m = 20$ and $m = 30$ latent variables. In addition, there is no guarantee that $\log(p(\mathbf{X}^*_{t_i}|\mathbf{X}_{t_i},\mathbf{X}_{t_{i+1}},\theta))$ is unimodal. It is likely that this method of generating proposals will perform poorly for SDEs with multi-modal marginal distributions.

It is worth adding that the description above is a slight simplification of the content of [55]. In fact, the authors suggest that $\mathbf{X}^*_{t_i}$ be broken into sub-blocks of random size and then updated. This has the effect of increasing the acceptance rate of the Metropolis-Hastings step, but it also increases the autocorrelation of the Markov chain.

In order to update θ, the authors use the factorisation

$$\begin{aligned} p(\theta|\mathbf{X}_{t_{1:n}},\mathbf{X}^*_{t_{1:n}}) &\propto p(\mathbf{X}_{t_{1:n}},\mathbf{X}^*_{t_{1:n}}|\theta)p(\theta) \\ &= \prod_{i=1}^{n}\prod_{j=0}^{M} p(\mathbf{X}_{t_{i,j+1}}|\mathbf{X}_{t_{i,j}},\theta) \end{aligned} \tag{4.6}$$

together with the Gaussian approximation (4.4). Here, we use the convention that $\mathbf{X}_{t_{i,0}} = \mathbf{X}_{t_i}$ and $\mathbf{X}_{t_{i,M+1}} = \mathbf{X}_{t_{i+1}}$. Equation (4.6) shows how to evaluate the conditional distribution for θ up to proportionality, so that a general Metropolis-Hastings update can be applied. The authors do not recommend a general update method for θ since it is usually model-dependent.

There is an important point in this paper that the authors have not explicitly considered, but which is hinted at in the numerical experiments. The problem was first described in Roberts and Stramer [57]. The issue is that in some cases, the imputed data can be highly informative about the value of a given parameter. When this is the case, only a small range of values of θ are consistent with the augmented data. $P(\theta|\mathbf{X}_{t_1},\mathbf{X}^*_{t_1},\mathbf{X}_{t_2},\mathbf{X}^*_{t_2},\dots)$ is tightly constrained, whereas $p(\theta|\mathbf{X}_{t_1},\mathbf{X}_{t_2},\dots)$ need not be. Proposed new values for θ are only accepted if they are close to current values of theta, and the Gibbs sampler only explores the space very slowly.

To see how this can happen in practice, consider the very simple case

$$d\mathbf{X}_t = \mathbf{b}d\mathbf{W}_t, \tag{4.7}$$

with an observation at $t = 0$ and $t = 1$. Imputed data points are introduced at times $(1/M, 2/M, \ldots, (M-1)/M)$. We then have

$$\begin{aligned} P(\mathbf{b}|\mathbf{X}_0, \mathbf{X}_{1/M}, \ldots, \mathbf{X}_1) &\propto P(\mathbf{X}_0, \mathbf{X}_{1/M}, \ldots, \mathbf{X}_1|\mathbf{b})P(\mathbf{b}) \\ &= p(\mathbf{b}) \prod_{i=1}^{M} \mathcal{N}(\mathbf{X}_{i/M}|\mathbf{X}_{(i-1)/M}, \mathbf{b}^2/M). \end{aligned} \tag{4.8}$$

Thus, the random variables $\{(\mathbf{X}_{i/M} - \mathbf{X}_{(i-1)/M})\}$ are i.i.d samples with from a normal distribution with mean 0 and variance b^2/M.

When M is large, the law of large numbers implies that

$$\frac{\mathbf{b}^2}{M} \approx \frac{1}{M} \sum_{i=1}^{M} (\mathbf{X}_{i/M} - \mathbf{X}_{(i-1)/M})^2. \tag{4.9}$$

Indeed, this is true with probability 1 as $M \to \infty$. This shows us that the sample path of $\mathbf{X}$ completely determines the value of $\mathbf{b}$. We can conclude that the conditional distribution of $\mathbf{b}$ is highly concentrated around the right-hand side of (4.9) when M is large.

Roberts and Stramer [57] suggest a solution in the one-dimensional setting with constant diffusion coefficient. The diffusion is transformed via

$$\dot{\mathbf{X}}_t := \frac{\mathbf{X}_t}{\mathbf{b}} \tag{4.10}$$

$$\ddot{\mathbf{X}}_t := \dot{\mathbf{X}}_t + \frac{(t_i - t)\dot{\mathbf{X}}_{t_{i-1}} + (t - t_{i-1})\dot{\mathbf{X}}_{t_i}}{t_i - t_{i-1}}, \qquad t_{i-1} \le t \le t_i. \tag{4.11}$$

The first transformation has the effect of 'moving the diffusion coefficient into the drift', while the second transformation means proposals can be generated using any tractable 'bridge' process that starts and ends at 0. For example, imputed data can be proposed by sampling from a Brownian bridge between these times.

The thesis of Kalogeropoulos [58] extends these ideas to the multivariate, partially observed setting where the diffusion coefficient may be dependent on the state. The transform (4.10) is replaced with

$$f(x) := \int_c^x \mathbf{b}^{-1}(u)du, \tag{4.12}$$

$$\dot{\mathbf{X}}_t := f(\mathbf{X}_t). \tag{4.13}$$

where c is an arbitrary constant. This is known as the *Lamperti transform*. It has the effect of setting the diffusion coefficient of the transformed diffusion $f(\mathbf{X})$ to the identity matrix. It is a critical component of a number of inference algorithms (see, for example, [59], [44]). However, the transform may not always exist (for example, when $\mathbf{b}$ is not invertible). When it does exist, it may be difficult to compute in closed form.

Golightly and Wilkinson [60] suggest a method for working around the problem of highly informative imputed data without the need to apply the Lamperti transform. They apply a 'whitening' step to the data before running the Gibbs sampler on θ and the augmented data. The signal is assumed to follow a multivariate, partially observed diffusion process with state-dependent diffusion coefficient. We will describe the technique applied to a process observed at times 0 and 1 with M imputed data points. The Euler discretisation of this process is given by

$$\mathbf{X}_{(i+1)/M} = \mathbf{X}_{i/M} + \mathbf{a}_\theta(\mathbf{X}_{i/M})\frac{1}{M} + \sqrt{\frac{1}{M}}\mathbf{b}_\theta(\mathbf{X}_{i/M})\mathbf{Z}_i, \tag{4.14}$$

where $\mathbf{Z}_i \sim \mathcal{N}(0,1)$. Given a set of imputed data and a fixed value of θ, one can deduce the value of the variables $\{\mathbf{Z}_i\}$ by re-arranging (4.14). Under the assumption that $\mathbf{b}$ is invertible,

$$\mathbf{Z}_i = \sqrt{M}\left(\mathbf{b}_\theta(\mathbf{X}_{i/M})\right)^{-1}\left(\mathbf{X}_{(i+1)/M} - \mathbf{X}_{i/M} - \mathbf{a}_\theta(\mathbf{X}_{i/M})\frac{1}{M}\right). \tag{4.15}$$

The authors suggest using a Gibbs sampler to alternate between sampling $\{\mathbf{Z}_i\}$ given θ, and sampling from θ given $\{\mathbf{Z}_i\}$. When running the Gibbs sampler, it is necessary to ensure that the simulations are always consistent with the data. This is achieved by adding a Metropolis-Hastings step, in which proposals are made from a 'bridge process' that satisfies

$$d\bar{\mathbf{X}}_t = \bar{a}(\bar{\mathbf{X}}_t)dt + \mathbf{b}_\theta(\bar{\mathbf{X}}_t)d\mathbf{W}_t, \tag{4.16}$$

where

$$\Sigma(x) = \mathbf{b}_\theta(x)\mathbf{b}_\theta(x)^\top \tag{4.17}$$

$$\bar{a}(x) = a_\theta(x) + \Sigma(x)\left(\Sigma(x)(1-t) + \mathbf{R}\right)^{-1}\left(\mathbf{Y}_1 - x - a_\theta(x)(1-t)\right). \tag{4.18}$$

The acceptance ratio for the proposals is computed by means of Girsanov's theorem (see Section 2.2.7). As t approaches 1, the second term on the right in (4.18) dominates, forcing $\bar{\mathbf{X}}$ towards $\mathbf{Y}_1$. This ensures the sample path ends up in the neighbourhood of the observation. When the covariance $\mathbf{R}$ of the observation noise is small and $\mathbf{b}$ is constant, the dynamics of the proposal match those of a Brownian bridge. This may

be a problem, as the dynamics of the original SDE may be very dissimilar to those of a Brownian bridge, resulting in a high rejection rate. We explore this possibility further in Chapter 6.

Perhaps the most sophisticated use of MCMC-based parameter estimation to date was developed in the series of papers [61, 62, 63, 64], (and applied in a different context in [44, 45]). The methodology is only applicable to diffusions that have a 'gradient drift'. That is, there exists a function $A : \mathbb{R}^n \to \mathbb{R}$ such that the drift $\mathbf{a}$ of the diffusion satisfies $\mathbf{a} = \nabla A$. These properties imply that the diffusion is reversible, and that the dynamics describe a random walk in a potential field. The diffusion coefficient must also satisfy $\mathbf{b}(x) = \mathbf{I}_n$. When this is not the case, one can sometimes transform $\mathbf{X}$ into the required form via the Lamperti transform. The assumptions are somewhat restrictive for multivariate diffusions, though one-dimensional diffusions are usually of the required form.

The authors describe a method that allows them to sample from certain classes of diffusion process with *no discretisation bias*. The sampling algorithm used by the authors is a form of rejection sampling which they call the 'exact algorithm'. The key idea behind the exact algorithm is that it is possible to decide whether to accept or reject a proposed path after inspecting it at a finite (albeit random) number of times. These times are the arrival times of a certain auxiliary Poisson process, and will be denoted $\{\psi_1, \ldots, \psi_k\}$, where k follows a Poisson distribution.

In order to construct a diffusion path starting at a point x and ending at y after a time t, the authors consider proposals from a Brownian bridge $\mathbf{B}$ starting at x and ending at y t units of time later. Values of the Brownian bridge $\{\mathbf{B}_{\psi_1}, \ldots, \mathbf{B}_{\psi_k}\}$ are sampled at times $\{\psi_1, \ldots, \psi_k\}$. The resulting collection of times and Brownian bridge values is known as the *skeleton*. The skeleton is accepted with a certain probability that depends on x, y, t and the drift function $\mathbf{a}$.

Once the skeleton has been accepted, one can sample from the diffusion $\mathbf{X}$ at any other time s by sampling from a Brownian bridge that is conditioned to hit the skeleton points at the appropriate times. This is computationally inexpensive as a result of the Markov property of Brownian bridges.

The authors use a Gibbs sampler to alternate between drawing from the skeleton process and drawing from the parameters θ that govern the drift function.

Andrieu et al. [65] describe a very general MCMC method for parameter estimation that is based on sequential importance sampling. The methodology, known as 'particle MCMC', builds on ideas that were first developed in [66]. The version

that is applicable to the estimation problem at hand is known as 'particle marginal Metropolis-Hastings' (PMMH). The idea itself is quite simple – the bulk of the paper is dedicated to proving correctness of the algorithm, which is not straightforward.

The basic idea is to replace the intractable quantities that one encounters in the parameter estimation problem with estimates obtained from a particle filter. The sampler targets the distribution

$$\begin{aligned}\pi(\mathbf{X}_{t_{1:n}}, \theta) = p(\mathbf{X}_{t_{1:n}}, \theta | \mathbf{Y}_{t_{1:n}}) &= p(\mathbf{X}_{t_{1:n}} \mid \theta, \mathbf{Y}_{t_{1:n}}) p(\theta | \mathbf{Y}_{t_{1:n}}) \\ &= p(\mathbf{X}_{t_{1:n}} | \theta, \mathbf{Y}_{t_{1:n}}) \frac{p(\mathbf{Y}_{t_{1:n}} \mid \theta) p(\theta)}{p(\mathbf{Y}_{t_{1:n}})} \end{aligned} \tag{4.19}$$

Samples from the conditional distribution of $\mathbf{X}$ can then be discarded, leaving a point estimate of $p(\theta | \mathbf{Y}_{t_{1:n}})$. The proposal distribution is chosen to be of the form

$$q\left((\theta, \mathbf{X}_{t_{1:n}}) \to (\theta^*, \mathbf{X}^*_{t_{1:n}})\right) = q(\theta^* | \theta) p(\mathbf{X}^*_{t_{1:n}} | \theta^*, \mathbf{Y}_{t_{1:n}}). \tag{4.20}$$

Here, $q(\theta^* | \theta)$ is a suitable Metropolis-Hastings proposal. The acceptance ratio is then

$$a\left((\theta, \mathbf{X}_{t_{1:n}}) \to (\theta^*, \mathbf{X}^*_{t_{1:n}})\right) = 1 \wedge \frac{p(\mathbf{Y}_{t_{1:n}} | \theta^*) p(\theta^*)}{p(\mathbf{Y}_{t_{1:n}} | \theta) p(\theta)} \frac{q(\theta | \theta^*)}{q(\theta^* | \theta)}, \tag{4.21}$$

since the term $p(\mathbf{X}^*_{t_{1:n}} | \theta^*, \mathbf{Y}_{t_{1:n}})$ in the proposal cancels with the same term in the target distribution. The only term in this expression that is difficult to estimate is $p(\mathbf{Y}_{t_{1:n}} | \theta^*)$. However, as we saw in Section 3.3.1, it is straightforward to estimate this term using sequential Monte-Carlo methods. The main drawback to this method is that it is expensive. One must run a new particle filter for each draw of θ^*.

4.2 Maximum likelihood estimation

In some situations, one may not need to compute the full Bayesian posterior distribution over parameters θ that govern a diffusion process. It might be sufficient to compute the maximum likelihood value of θ.

In the case of error-free observations of a process, this is equivalent to finding the value of θ that maximises

$$\log\left(p(X_{t_{1:n}} | \theta)\right) = \sum_{i=1}^{n-1} \log\left(p(X_{t_{i+1}} | X_{t_i}, \theta)\right). \tag{4.22}$$

In this simple case, the first problem one encounters is to find an efficient way of estimating $\log\left(p(X_{t_{i+1}} | X_{t_i}, \theta)\right)$. This is the issue that is addressed in Durham and Gallant

[67], who study the univariate error-free problem. In the examples considered in the paper, the parameter vector is low-dimensional, and it appears that the log-likelihood (4.22) has been evaluated on a grid of discrete values of θ. Clearly, such a method would not scale beyond a handful of dimensions, but in some contexts it is of interest.

One can make the case that [67] is largely a review paper, though it also makes some novel contributions. The authors point out that one can use a Brownian bridge together with an application of Girsanov's theorem in order to construct an importance sampler for use in a Monte-Carlo estimate of the transition density. They also show how to modify the Brownian bridge proposals so that the resulting estimates are more stable.

Archambeau et al. [68, 24, 69] define a notion of Kullback-Leibler (KL) divergence between two diffusion processes (or more accurately, two distributions over function space that are defined by diffusion processes). In order for the KL divergence to be finite, the diffusion processes must have identical diffusion coefficient, which must be state-independent. We define these processes as

$$d\mathbf{X}_t = \mathbf{a}(\mathbf{X}_t)dt + \mathbf{b}d\mathbf{W}_t, \tag{4.23}$$

$$d\bar{\mathbf{X}}_t = \bar{\mathbf{a}}(\bar{\mathbf{X}}_t)dt + \mathbf{b}d\mathbf{W}_t. \tag{4.24}$$

Let P be the distribution on $\mathbf{X}$ (which is a distribution on the space of continuous functions) and let Q be the the distribution for $\bar{\mathbf{X}}$, so that for any set $\mathcal{S}$ of functions,

$$p(\mathbf{X} \in \mathcal{S}) = \int_S dP(\omega), \tag{4.25}$$

and similarly for Q and $\bar{\mathbf{X}}$. According to Girsanov's theorem, the 'density ratio' (or more accurately, the Radon-Nikodym derivative) between two diffusion processes is

$$\frac{dP}{dQ} = \exp\left(\int_0^T \left(\frac{\mathbf{a}(\mathbf{X}_u) - \bar{\mathbf{a}}(\mathbf{X}_u)}{\mathbf{b}}\right) dW_u - \frac{1}{2}\int_0^T \left(\frac{\mathbf{a}(\mathbf{X}_u) - \bar{\mathbf{a}}(\mathbf{X}_u)}{\mathbf{b}}\right)^2 du\right), \tag{4.26}$$

where T is some timescale of interest. See Section 2.2.7 for more details and an informal derivation. We take the expectation of the logarithm of (4.26) (recalling that Ito integrals have zero mean) to see that

$$\mathrm{KL}(Q||P) = \frac{1}{2}\int_0^T \mathbb{E}_Q\left[\left(\frac{\mathbf{a}(\mathbf{X}_u) - \bar{\mathbf{a}}(\mathbf{X}_u)}{\mathbf{b}}\right)^2\right] du, \tag{4.27}$$

where $\mathbb{E}_Q[\cdot]$ is understood to be 'expectation with respect to the distribution Q'. Note that the authors derived (4.27) by discretising the processes and passing to the continuum limit rather than by applying Girsanov's theorem. However, it is entirely likely

that the authors are aware of the derivation given here, and indeed the derivation via discretisation is arguably more suitable for an audience of non-specialists. The derivation is similar for vector-valued processes, with the KL divergence given by

$$\mathrm{KL}(Q||P) = \int_0^T \mathbb{E}_Q \left[(\mathbf{a}(\mathbf{X}_u) - \bar{\mathbf{a}}(\mathbf{X}_u))^\top (\mathbf{b}\mathbf{b}^\top)^{-1} (\mathbf{a}(\mathbf{X}_u) - \bar{\mathbf{a}}(\mathbf{X}_u)) \right] du \tag{4.28}$$

The authors formulate the inference problem as a constrained optimisation problem. The aim is to minimise the vatiational free energy

$$\mathcal{F}(Q,\theta) = -\mathbb{E}_Q \left[\log \left(\frac{p(\mathbf{X}, \mathbf{Y}_{t_{1:n}}|\theta)}{Q(\mathbf{X})} \right) \right]. \tag{4.29}$$

This is equivalent to minimising the KL divergence. The optimisation problem is used to find the process with *linear, time-dependent drift* that minimises the free energy. Statistics of this new process are tractable, and so inference is performed on the variational approximation.

Aït-Sahalia [70, 71, 59] estimates the transition density $p(\mathbf{X}_{t_i}|\mathbf{X}_{t_{i-1}}, \theta)$ (and hence the log-likelihood, via (4.22)) by means of an expansion in *Hermite polynomials*. These polynomials form an orthonormal basis of the Hilbert space whose inner product is given by

$$\langle f, g \rangle_H = \frac{1}{\sqrt{2\pi}} \int f(u) g(u) e^{-u^2} du. \tag{4.30}$$

In fact, the Hermite polynomials are the result of performing the Gram-Schmidt procedure on $\{1, x, x^2, \dots\}$ with the inner product (4.30).

Aït-Sahalia first transforms the diffusion into the form

$$d\mathbf{X}_t = \mathbf{a}(\mathbf{X}_t)dt + d\mathbf{W}_t, \qquad \mathbf{X}_0 = 0. \tag{4.31}$$

This is further rescaled via $\hat{\mathbf{X}}_{t_i} = \mathbf{X}_{t_i}/\sqrt{(t_i - t_{i-1})}$, so that $p(\hat{\mathbf{X}}_{t_i}|\hat{\mathbf{X}}_{t_{i-1}})$ resembles a standard normal distribution as closely as possible.

To illustrate Aït-Sahalia's approach, we introduce alternative notation for the transition density:

$$f(x, y, \Delta t) = \lim_{\delta x \to 0} \mathbb{P}(\hat{\mathbf{X}}_{t_i} \in \delta x | \hat{\mathbf{X}}_{t_{i-1}} = y)/\delta x, \tag{4.32}$$

where $\Delta t = t_{i+1} - t_i$. We can expand the left-hand side of (4.32) in terms of Hermite polynomials in x:

$$f(x, y, \Delta t) = \frac{e^{x^2}}{\sqrt{2\pi}} \sum_{i=1}^{\infty} \langle f(\cdot, y, \Delta t), H_i \rangle H_i(x). \tag{4.33}$$

Note that we use the standard inner product rather than (4.30). The transition density can be approximated by truncating the sum in (4.33). All that remains is to estimate

$\langle f(\cdot, y, \Delta t), H_i \rangle$. It turns out this is equivalent to estimating $\mathbb{E}[H(\mathbf{X}_{t_i})|\mathbf{X}_{t_{i-1}}]$. Aït-Sahalia does this by truncating a Taylor expansion in Δt. This 'Taylor' expansion is actually the stochastic analytic counterpart of the ordinary Taylor expansion (making use of the so-called infinitesimal generator of the process), but the idea is the same. The result is asymptotically exact as the number of terms in the Taylor series and the number of terms in the truncation of (4.33) tend to infinity.

One practical drawback of the method is that the formulas are unweildy. For example, the third-order Taylor expansion of $\langle f(\cdot, y, \Delta t), H_1 \rangle$ contains eight terms, which involve up to fourth-order derivatives of $\mathbf{a}$. Indeed, multiple pages of [71] are dedicated to listing formulas for the coefficients in a simple two-dimensional example.

Numerical experiments in [70] and [59] are unfortunately limited. The papers do not explore performance of the method on a dataset generated by a nonlinear diffusion, though details are provided for an Ornstein-Uhlenbeck process. As a result of the limited experiments and difficulty of implementation, it is unclear whether this method could be effective in practice.

Another intriguing way of proceeding comes from the exact sampling approach of Beskos et al. [72], which was discussed earlier in Section 4.1. As before, the method is only applicable to gradient diffusions – i.e., processes with unit diffusion coefficient and a drift that can be expressed as the gradient of a function. It turns out that the density of a gradient diffusion satisfies

$$\begin{aligned} p(\mathbf{X}_t|\mathbf{X}_0) =& \mathcal{N}(\mathbf{X}_t - \mathbf{X}_0|0, t\mathbf{I}_n) \exp\left(A(\mathbf{X}_t) - A(\mathbf{X}_0)\right) \\ &\times \mathbb{E}\left[\exp\left(-\int_0^t \phi(\mathbf{B}_s)ds\right)\right], \end{aligned} \tag{4.34}$$

where $\phi(x) = \|\mathbf{a}(x)\|^2 + \nabla^2 A(x)$, and the expectation is over all Brownian bridges $\mathbf{B}$ starting at $\mathbf{X}_0$ and ending at $\mathbf{X}_t$. These terms are all tractable aside from the expectation on the right. However, it is possible to construct an unbiased estimator of this quantity by sampling from a certain Poisson process and computing statistics of that process. Moreover, one can use this technique to obtain an estimate of the likelihood function as a whole, rather than pointwise estimates for specified values of θ.

The ability to construct an unbiased estimate of the transition density has uses beyond maximum likelihood estimation. In [44], [45], these ideas were used to construct a particle filter in which the weights are unbiased estimates of the 'true' weights. The work is discussed further in Section 8.

Part II

Research

Chapter 5

The series expansion approximation and SDEs

In this chapter, we introduce a 'series expansion approximation' to a stochastic differential equation. The basic insight behing the approximation is that Brownian motion can be 'simplified' in some sense, by throwing away the high-frequency components of the process. The remaining process can be described using a relatively small number of covariates. We will use the 'simplified' Brownian motion as the driving noise in an approximation to a stochastic differential equation.

In Section 5.1.1, we introduce the series expansion approximation. Issues surrounding convergence of the approximation to the true process are discussed in Section 5.1.2. In Section 5.1.3, we show that it is possible in some cases to set the approximation error to 0 at a given time. In Section 5.1.4, we investigate the accuracy of the approximation with a number of numerical experiments. We show how the approximation can fail in 'resonant' systems in Section 5.1.5.

5.1 Brownian Series expansions

In Section 2.1, we described a construction of Brownian motion in terms of an orthonormal basis of $L^2[0,T]$, where T is some timescale of interest. We will now re-visit that construction. We will show how it can be used to generate approximate sample paths from a diffusion process. We will briefly discuss how this approximation can be exploited for inference and parameter estimation. This discussion will be expanded upon in Chapters 6, 7 and 8.

We will begin with the same set-up as in Section 2.1. That is, we choose an or-

thonormal basis $\{\phi_i\}$ of $L^2[0,T]$. We first present an alternate version of the derivation, which uses Ito calculus. This derivation, adapted from [73], provides more insight into the role of the random coefficients $\{\mathbf{Z}_i\}$ that appear in (2.25).

Suppose $\mathbf{W} = (W^{(1)}, \ldots, W^{(d)})$ is a standard d-dimensional Brownian motion, and let $\{\phi_i\}_{i\geq 1}$ be an orthonormal basis of $L^2([0,T],\mathbb{R})$. We use the notation $\mathbb{I}_{\{A\}}(\cdot)$ to denote the indicator function. That is, $\mathbb{I}_{\{[0,t]\}}(u) = 1$ when $0 \leq u \leq t$, and $\mathbb{I}_{\{[0,t]\}}(u) = 0$ otherwise. One can construct a series expansion of $\mathbf{W}$ in terms of the basis functions $\{\phi_i\}$ as follows:

$$\begin{aligned}
\mathbf{W}_t &= \int_0^T \mathbb{I}_{\{[0,t]\}}(u) d\mathbf{W}_u \\
&= \int_0^T \left(\sum_{i=1}^{\infty} \langle \mathbb{I}_{\{[0,t]\}}, \phi_i \rangle \phi_i(u) \right) d\mathbf{W}_u \\
&= \sum_{i=1}^{\infty} \left(\int_0^T \phi_i(u) d\mathbf{W}_u \right) \langle \mathbb{I}_{\{[0,t]\}}, \phi_i \rangle \\
&= \sum_{i=1}^{\infty} \left(\int_0^T \phi_i(u) d\mathbf{W}_u \right) \int_0^t \phi_i(u) du. \qquad (5.1)
\end{aligned}$$

We use the standard inner product on $L^2[0,T]$, which is defined as

$$\langle f, g \rangle = \int_0^T f(u) g(u) du. \qquad (5.2)$$

For ease of notation, we set

$$\mathbf{Z}_i = \int_0^T \phi_i(u) d\mathbf{W}_u. \qquad (5.3)$$

We can see that the stochastic integrals are i.i.d d-dimensional standard normal by noting that the basis functions are deterministic (so that the integrals follow a normal distribution). Ito integrals have mean 0, so that

$$\mathbb{E}[\mathbf{Z}_i] = 0. \qquad (5.4)$$

Finally, by Ito's isometry,

$$\begin{aligned}
\mathrm{Cov}(\mathbf{Z}_i, \mathbf{Z}_j) &= \mathbb{E}\left[\left(\int_0^T \phi_i(u) d\mathbf{W}_u \right) \left(\int_0^T \phi_j(u) d\mathbf{W}_u \right)^\top \right] \\
&= \left(\int_0^T \phi_i(u) \phi_j(u) du \right) \mathbf{I}_d = \delta_{ij} \mathbf{I}_d. \qquad (5.5)
\end{aligned}$$

Here, $\mathbf{I}_d$ is the $d \times d$ identity matrix.

We conclude that

$$\mathbf{W}_t = \sum_{i=1}^{\infty} \mathbf{Z}_i \int_0^t \phi_i(u) du. \qquad (5.6)$$

We can obtain an approximation of a Brownian sample path by drawing i.i.d samples $\mathbf{Z}_i$ from a standard normal distribution and truncating the sum in (5.6). This allows us to describe a Brownian sample path approximately in terms of a finite number of variates.

We will adopt the convention that the basis functions are ordered by the number of times they change sign on the interval $[0,T]$. Thus, when i is small, $\mathbf{Z}_i$ governs low-frequency oscillations of the Brownian motion. We will see that in most cases, the most 'interesting' coefficients will be those that govern the low-frequency terms.

5.1.1 Series Expansion Approximation of SDE

The heuristic behind the series expansion approximation is that high-frequency, low-amplitude oscillations in (5.6) should 'matter' less than the low-frequency, high amplitude terms in the Brownian motion when it is used as the driving noise in a diffusion process. This is because high-frequency terms change sign frequently. A system driven by a high-frequency signal will be 'pushed' in one direction for a short time. When the driving signal changes sign, this will create a 'push' in the opposite direction. This second force will often mitigate the effect of the first. We do not expect that the heuristic will work in all situations: for example, the reasoning we have outlined here is obviously not applicable to resonant systems.

We now present a brief description of the series expansion approximation. As usual, we will assume the dynamics of the diffusion satisfy

$$d\mathbf{X}_t = \mathbf{a}(\mathbf{X}_t)dt + \mathbf{b}(\mathbf{X}_t)d\mathbf{W}_t \qquad \mathbf{X}_0 = x_0. \tag{5.7}$$

According to this heuristic, we should be able to truncate (5.6) after, say, N terms without inducing significant bias in the distribution of $\mathbf{X}$ at some time $t \leq T$. Of course, the nature of this bias (and therefore the choice of N) depends on the choice of $\{\phi_i\}$ and the dynamics of the SDE.

We can formally differentiate both sides of (5.6), yielding

$$\frac{d\mathbf{W}_t}{dt} = \sum_{i=1}^{\infty} \mathbf{Z}_i \phi_i(t). \tag{5.8}$$

Note that equation (5.8) is strictly formal: the sum on the right diverges with probability 1, reflecting the fact that Brownian motion is nowhere differentiable. Truncating the sum gives the approximation

$$d\mathbf{W}_t \approx \sum_{i=1}^{N} \mathbf{Z}_i \phi_i(t) dt. \tag{5.9}$$

We can substitute this approximation for $d\mathbf{W}_t$ in (5.7). We will refer to this new approximate process as $\hat{\mathbf{X}}$. Since $\hat{\mathbf{X}}$ is driven by a finite linear combination of basis functions, the resulting process is differentiable. We can therefore interpret $\hat{\mathbf{X}}$ as the solution to an ordinary differential equation, which satisfies

$$\frac{d\hat{\mathbf{X}}_t}{dt} = \mathbf{a}(\hat{\mathbf{X}}_t) + \mathbf{b}(\hat{\mathbf{X}}_t)\sum_{i=1}^{N}\mathbf{Z}_i\phi_i(t). \tag{5.10}$$

The approximation (5.10) has the advantage of re-casting an infinite dimensional problem in finite-dimensional terms. Given the value of $\hat{\mathbf{X}}_{t_{k-1}}$ as an initial condition, we can view the solution of (5.10) as a function

$$\hat{\mathbf{X}}_{t_k} = f(T, \hat{\mathbf{X}}_{t_{k-1}}, \mathbf{Z}_{1:N}), \tag{5.11}$$

where $T = t_k - t_{k-1}$. Here, f solves the ordinary differential equation (5.10), and $\{\mathbf{Z}_i\}_{1\le i\le N}$ are i.i.d standard d-dimensional Gaussian random variables. In essence, the time-t distribution of the process $\hat{\mathbf{X}}$ can be interpreted as the image of a Gaussian distribution under a nonlinear transform.

Of course, one can apply this reasoning to other methods for approximating SDEs. In the N-step Euler-Maruyama scheme, one sets $\Delta t = T/N$, and

$$\mathbf{X}_{(n+1)\Delta t} = g(\mathbf{X}_{(n\Delta t}, \mathbf{Z}_n) = \mathbf{X}_{n\Delta t} + \mathbf{a}(\mathbf{X}_{n\Delta t})\Delta t + \mathbf{b}(\mathbf{X}_{n\Delta t})\mathbf{Z}_n\sqrt{\Delta t}. \tag{5.12}$$

The equivalent of f in (5.11) is the N-fold application of g to the initial state $\mathbf{X}_0$. However, this approximation puts all random variables $\{\mathbf{Z}_i\}$ on an 'equal footing', and loses the interpretation of the coefficients as controllers of the behaviour of the SDE at varying frequencies.

5.1.2 Convergence of the series expansion method

We now discuss asymptotic convegence of the approximation (5.10) to the solution of the true SDE. Wong and Zakai [74] showed that in the univariate case, under mild technical assumptions, the solution of (5.10) converges to the Stratonovich solution of the SDE that is being approximated.

We use the circle notation to denote Stratonovich integration. Recall that a Stratonovich SDE

$$d\mathbf{X}_t = \mathbf{a}(\mathbf{X}_t)dt + \mathbf{b}(\mathbf{X}_t)\circ d\mathbf{W}_t \tag{5.13}$$

can be converted to an Itô SDE and vice versa using the relationship

$$\int_0^t \mathbf{b}(\mathbf{X}_t)\circ d\mathbf{W}_t = \int_0^t \mathbf{b}(\mathbf{X}_t)d\mathbf{W}_t + \int_0^t \mathbf{c}(\mathbf{X}_t)dt, \tag{5.14}$$

where the integral on the left is in the Stratonovich sense, and the i-th component of the vector $\mathbf{c}$ satisfies

$$\mathbf{c}^i(x) = -\frac{1}{2}\sum_{j=1}^{n}\sum_{k=1}^{d}\mathbf{b}^{j,k}(x)\frac{\partial \mathbf{b}^{i,k}}{\partial x_j}(x). \tag{5.15}$$

In other words, the Stratonovich solution of an SDE is equivalent to the Itô solution with a modified drift.

The issue of convergence in the multidimensional setting is more complicated than in the univariate case, as was observed by McShane [75] (see also [76], [77]). In general, if $\{\mathbf{W}_n\}$ is a sequence of piecewise smooth processes converging to a Brownian motion, one cannot guarantee $\{\mathbf{W}_n\} \to \mathbf{W}$ implies that the sequence of approximate differential equations converges to the Stratonovich solution of the SDE. See McShane [75], Section 10 for a concrete counterexample in two dimensions.

However, if one chooses the Haar wavelets (described in Section 5.1.3 below) as an orthonormal basis in which to expand the driving Brownian motion, then convergence is guaranteed. In practical tests, we have not observed failure of the series expansion to converge to the Stratonovich limit when other orthonormal bases of $L^2[0,T]$ are used.

5.1.3 Exact solutions

It turns out that one can often choose an appropriate set of basis functions so that the error vanishes at a specific time T. We will elaborate on this point, but first we introduce an inportant orthonormal basis of $L^2[0,T]$: the Haar wavelets. The Haar wavelets are parametrised by two natural numbers: the scale, $n \geq 0$, and the shift $0 \leq k < 2^n$. The first wavelet is defined as

$$\psi_{0,0}(t) = \begin{cases} 1 & 0 \leq t < \frac{T}{2} \\ -1 & \frac{T}{2} < t \leq T \\ 0 & \text{otherwise.} \end{cases} \tag{5.16}$$

Further wavelets are defined by rescaling $\psi_{0,0}$ so that it is non-zero only on some subinterval of $[0,T]$ while ensuring that the wavelet still has norm 1. In general,

$$\psi_{n,k}(t) = \frac{2^{n/2}}{\sqrt{T}}\psi_{0,0}\left(2^n t - kT\right), \qquad 0 \leq k < 2^n. \tag{5.17}$$

Thus, $\psi_{1,0}$ is a 'copy' of $\psi_{0,0}$ which has been rescaled and restricted to $[0,T/2]$, and $\psi_{1,1}$ is a 'copy' restricted to $[T/2,T]$. We also add the constant function $\psi_*(t) = 1/\sqrt{T}$ to the set to form a complete basis.

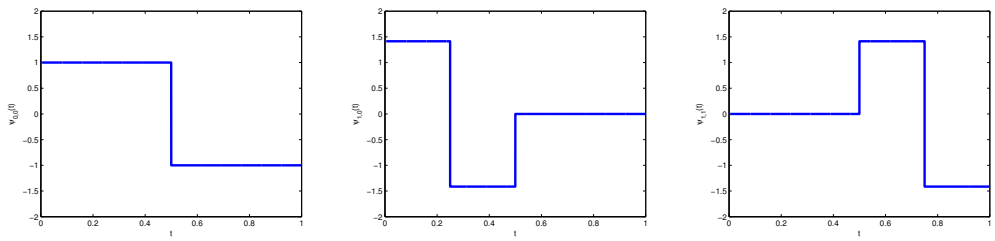


Figure 5.1: The Haar wavelets $\psi_{0,0}$, $\psi_{1,0}$ and $\psi_{1,1}$.

To be consistent with the notation of Section 5.1.1, we set $\phi_1 = \psi_*$, $\phi_2 = \psi_{0,0}$, $\phi_3 = \psi_{1,0}$, and so on. Note that, by (5.3), $Z_1 = \mathbf{W}_T/\sqrt{T}$. This is an important property of any basis function expansion where ϕ_1 is constant.

We can now return to our discussion of the accuracy of the approximation. As a simple example of how the series expansion can be exact at one particular time, we will consider geometric Brownian motion, which is defined by the SDE

$$d\mathbf{X}_t = a\mathbf{X}_t dt + b\mathbf{X}_t \circ d\mathbf{W}_t, \qquad \mathbf{X}_0 = 1 \tag{5.18}$$

with $a, b \geq 0$. Recall that the circle notation means we are interpreting the SDE in the Stratonovich sense. At time T, the solution to this equation is

$$\mathbf{X}_T = \exp\left(aT + b\mathbf{W}_T\right). \tag{5.19}$$

This is identical to the time-T solution of the one-term series expansion approximation using the Haar wavelet basis, which is given by

$$\begin{aligned} d\mathbf{X}_t &= a\mathbf{X}_t dt + b\mathbf{X}_t \left(\mathbf{Z}_1 \phi_1(t)\right) dt \qquad \mathbf{X}_0 = 1 \\ &= a\mathbf{X}_t dt + b\mathbf{X}_t \left(\frac{\mathbf{W}_T}{T}\right) dt \end{aligned} \tag{5.20}$$

Similarly, the approximation is exact at times T and $T/2$ when using the basis functions ϕ_1 and ϕ_2, and so on. We expect a similar result to hold for any SDE that can be written in the form

$$dh^{-1}(\mathbf{X}_t) = d\mathbf{W}_t, \tag{5.21}$$

for some function h so that $\mathbf{X}_t = h(\mathbf{W}_t)$. That is, the solution at time t depends only on the value of $\mathbf{W}$ at time t. See [22], Chapter 4 for a large number of examples SDEs with this property.

A slightly more complicated example of exactness of the series expansion method at time T comes from considering the Ornstein-Uhlenbeck process

$$d\mathbf{X}_t = a\left(\theta - \mathbf{X}_t\right) dt + b d\mathbf{W}_t, \qquad \mathbf{X}_0 = x_0. \tag{5.22}$$

The solution to this SDE at time T is

$$\mathbf{X}_T = \mathbf{X}_0 e^{-aT} + \theta\left(1 - e^{-aT}\right) + \int_0^T b e^{a(u-T)} d\mathbf{W}_u. \tag{5.23}$$

A general N-term series expansion approximation (in some as-yet undetermined basis) of $\mathbf{X}$ takes the form

$$d\hat{\mathbf{X}}_t = a\left(\theta - \hat{\mathbf{X}}_t\right) dt + \sum_{i=1}^{N} b\mathbf{Z}_i \phi_i(t) dt, \qquad \hat{\mathbf{X}}_0 = x_0. \tag{5.24}$$

The solution of this equation at time T is

$$\hat{\mathbf{X}}_T = \hat{\mathbf{X}}_0 e^{-aT} + \theta\left(1 - e^{-aT}\right) + \sum_{i=1}^{N} \mathbf{Z}_i b \int_0^T e^{a(u-T)} \phi_i(u) du. \tag{5.25}$$

Note that each term in the sum on the right can be interpreted as an inner product on $L^2[0,T]$. We can choose ϕ_1 arbitrarily, so we set it to

$$\phi_1(t) = \frac{e^{a(t-T)}}{\|e^{a(\cdot - T)}\|}. \tag{5.26}$$

From (5.26),

$$\begin{aligned} \int_0^T e^{a(u-T)} \phi_i(u) du &= \|e^{a(\cdot - T)}\| \int_0^T \phi_1(u)\phi_i(u) du \\ &= \|e^{a(\cdot - T)}\| \langle \phi_1, \phi_i \rangle \\ &= 0 \qquad \forall i > 1, \end{aligned} \tag{5.27}$$

since the basis functions are orthogonal. Thus, at time T all terms except the first vanish in the sum in (5.25). Thus, with this choice of ϕ_1, a single term in the series expansion is sufficient to guarantee exactness of the approximation.

5.1.4 Accuracy of the approximation

In the general nonlinear case, analytic solutions for multi-dimensional ordinary differential equations are rarely available in closed form. Hence, it is difficult to establish precise bounds on the error induced by the series expansion approximation. In this section we aim to investigate properties of the series expansion approximation numerically. We will focus on the error induced at time T rather than error in the entire path, since the time-T error is more useful when considering inference problems involving discretely observed diffusions.

As a first example, we will consider the double-well process

$$d\mathbf{X}_t = 4\mathbf{X}_t(1-\mathbf{X}_t^2)dt + dW_t, \qquad \mathbf{X}_0 = 1. \tag{5.28}$$

at times $T = 1$ and $T = 2$. This is a bistable distribution that is often used as a test for numerical methods involving nonlinear SDEs [24, 69, 78, 79]. One could of course choose values for α, γ and T that would result in poor convergence. Empirically, we see that we need more terms in the series expansion as T grows large. We can see that for a fixed number of terms, the samples in Figure 5.2 (time $T = 1$) are closer to the true distribution than the samples in Figure 5.3 (time $T = 2$).

However, any diffusion process on an interval $[0, \kappa]$ can be rescaled to take values on the interval $[0, 1]$ only, via the transform $\mathbf{V}_t := \mathbf{X}_{t/\kappa}$. If $\mathbf{X}$ has dynamics

$$d\mathbf{X}_t = \mathbf{a}(\mathbf{X}_t)dt + \mathbf{b}(\mathbf{X}_t)d\mathbf{W}_t, \tag{5.29}$$

then the transformed process $\mathbf{V}$ has dynamics

$$d\mathbf{V}_t = \kappa\mathbf{a}(\mathbf{V}_t)dt + \sqrt{\kappa}\mathbf{b}(\mathbf{V}_t)d\mathbf{W}_t. \tag{5.30}$$

We conclude from Figures 5.2 and 5.3 that the series expansion method is more accurate for (5.28) than it is for the rescaled process

$$d\mathbf{X}_t = 8\mathbf{X}_t(1-\mathbf{X}_t)dt + \sqrt{2}dW_t, \qquad \mathbf{X}_0 = 1. \tag{5.31}$$

This analysis suggests that larger drift and diffusion coefficients may degrade the performance of the series expansion method. This may be explained by the fact that the variance at time T of the series expansion approximation is lower than that of the true process. This is a result of truncating the Brownian series expansion (5.6), which reduces the variance of the driving noise.

We now test the approximation on a model of an aircraft turning in the (x_1, x_3) plane. This model is used to test the algorithms in Sections 7 and 8. We model the motion of the aircraft using noisy dynamics that account for imperfections in the control system. The model also accounts for external forces such as wind that might affect the trajectory of the aircraft. We describe the state of the with a seven-dimensional vector $x_{1:7}$. The components (x_1, x_3, x_5) represent the position of the aircraft in rectangular Cartesian coordinates, while the components (x_2, x_4, x_6) describe its velocity. The number x_7 describes the rate at which the aircraft is turning in the (x_1, x_3) plane.

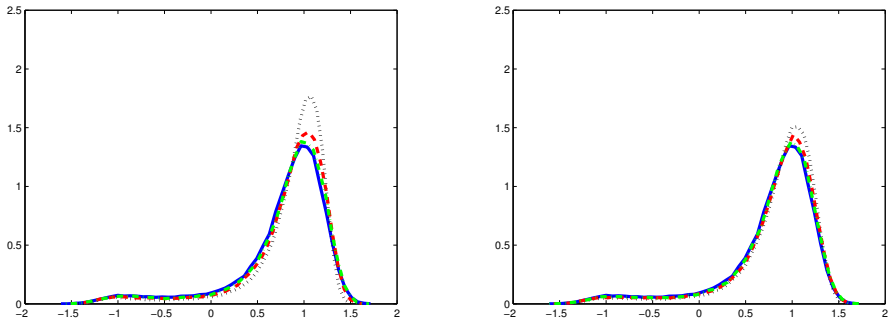


Figure 5.2: (Histogram of samples from a series expansion approximation of (5.28) at time $T = 1$ using the Haar wavelet basis (left) and Fourier Sine series (right). We used $N = 4$ terms (black dots), $N = 8$ terms (red dashes), and $N = 16$ terms (green dots and dashes). The blue line represents a very fine Euler-Maruyama discretisation ($\Delta t = .0001$), which we take as ground truth.

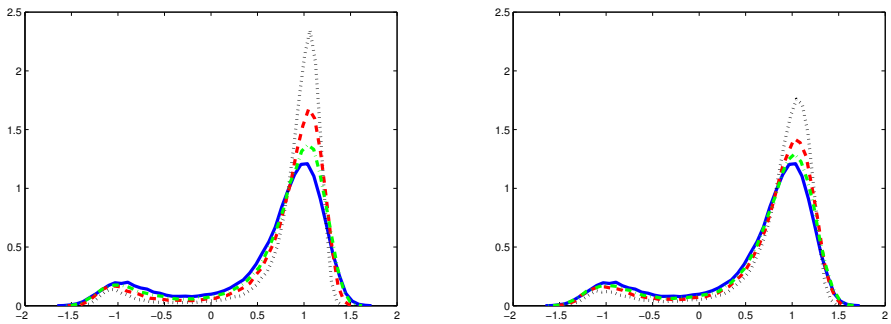


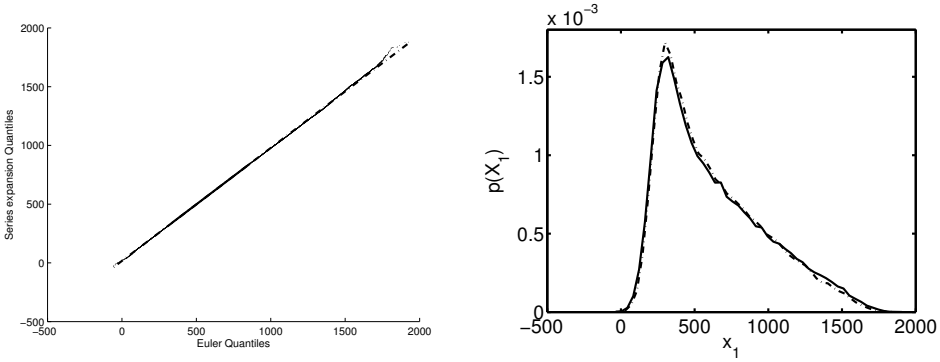
Figure 5.3: (Histogram of samples from a series expansion approximation of (5.28) at time $T = 2$ (or equivalently samples from (5.31) at time $T = 1$) using the Haar wavelet basis (left) and Fourier Sine series (right). We used $N = 4$ terms (black dots), $N = 8$ terms (red dashes), and $N = 16$ terms (green dots and dashes). The blue line represents a very fine Euler-Maruyama discretisation ($\Delta t = .0001$), which we take as ground truth.

The dynamics of the system are given by (5.29), with

$$\mathbf{a}(x_{1:7}) = \begin{pmatrix} x_2 \\ -x_7 x_4 \\ x_4 \\ x_7 x_2 \\ x_6 \\ 0 \\ 0 \end{pmatrix} \tag{5.32}$$

$$\mathbf{b}(x_{1:7}) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ \frac{\sqrt{1+x_2^2}}{v} & \frac{\sqrt{1+x_4^2}}{v_{xy}} & \frac{\sqrt{(1+x_2^2)(1+x_6^2)}}{vv_{xy}} & 0 \\ 0 & 0 & 0 & 0 \\ \frac{\sqrt{1+x_4^2}}{v} & -\frac{\sqrt{1+x_2^2}}{v_{xy}} & \frac{\sqrt{(1+x_4^2)(1+x_6^2)}}{vv_{xy}} & 0 \\ 0 & 0 & 0 & 0 \\ \frac{\sqrt{1+x_6^2}}{v} & 0 & -\frac{v_{xy}}{v} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{5.33}$$

Here, $v = \sqrt{1+x_2^2+x_4^2+x_6^2}$ and $v_{xy} = \sqrt{1+x_2^2+x_4^2}$. Nonlinearities arise from two sources in this system. Firstly, the state-dependent covariance matrix causes the system to deviate from Gaussianity. Second, the random evolution of the turn rate $\mathbf{X}_7(\cdot)$ causes the aircraft to behave erratically. As the variance of $\mathbf{X}_7(\cdot)$ grows, the system becomes more nonlinear and more non-Gaussian. A similar model was studied in [80], though in that case the diffusion matrix was assumed to be constant. Note that the state dependent covariance matrix makes Itô-Taylor and Runge-Kutta discretisations difficult to implement.



(a) a (b) b

Figure 5.4: (Top) Q-Q plot of 100,000 samples from an Euler-Maruyama discretisation of $\mathbf{X}_1(T=8)$ versus 100,000 samples from the series expansion approximation. Linearity of the plot suggests the distributions are very similar. (Bottom) Density plots of the samples. Draws from the Euler scheme are plotted using the solid line, and draws from the series expansion scheme are represented by the broken line. We used the Fourier sine series as a basis, with $N = 10$.

In order to test the series expansion approximation, we simulated paths from $\mathbf{X}$ on the interval $[0,8]$. We set $\mathbf{X}_0 = (1000, 0, 2650, 150, 200, 0, 6)$, and $\mathrm{Cov}(\mathbf{W}_t) =$

	Euler	N = 1	N = 4	N = 6	N = 10
$\mathbb{E}[\mathbf{X}_1(t)]$	626	549	607	612	619
$\mathbb{E}[\mathbf{X}_2(t)]$	-59	-91	-65	-63	-61
$\mathbb{E}[\mathbf{X}_3(t)]$	3588	3689	3612	3603	3597
$\mathbb{E}[\mathbf{X}_4(t)]$	53	82	58	56	55
$\mathbb{E}[\mathbf{X}_5(t)]$	200	200	200	200	200
$\mathbb{E}[\mathbf{X}_6(t)]$	0	0	0	0	0
$\mathbb{E}[\mathbf{X}_7(t)]$	6	6	5.9	6	5.9

Table 5.1: Marginal mean values for $\mathbf{X}_{1:7}(t=8)$ as computed by the Euler scheme and series expansion approximations

	Euler	N = 1	N = 4	N = 6	N = 10
$\mathrm{Std}(\mathbf{X}_1(t))$	359	151	317	333	346
$\mathrm{Std}(\mathbf{X}_2(t))$	90	61	86	88	89
$\mathrm{Std}(\mathbf{X}_3(t))$	277	128	250	261	268
$\mathrm{Std}(\mathbf{X}_4(t))$	93	66	90	91	92
$\mathrm{Std}(\mathbf{X}_5(t))$	29	17	27	28	28
$\mathrm{Std}(\mathbf{X}_6(t))$	6.4	5.7	6.2	6.3	6.3
$\mathrm{Std}(\mathbf{X}_7(t))$	14.1	12.8	13.7	13.9	14.0

Table 5.2: Marginal standard deviations for $\mathbf{X}_{1:7}(t=8)$ as computed by the Euler scheme and series expansion approximations

$\mathrm{Diag}(50,50,50,25)t$, resulting in a highly nonlinear process. We took $100,000$ simulations from the Euler-Maruyama scheme as ground truth, having set $\Delta t = .005$. The basis functions were defined by

$$\phi_k(t) = \sqrt{\frac{2}{T}} \sin\left(\frac{(k-\frac{1}{2})\pi t}{T}\right), \tag{5.34}$$

with $T = 8$. We simulated $100,000$ paths from the series expansion approximation with $N = 1,4,6$ and 10. The marginal means and standard deviations are shown in Tables 1 and 2. Figure 5.4 shows a Q-Q plot of the Euler simulation versus the series expansion simulation with $N = 10$, together with a plot of both densities.

When implementing a test such as this, one must consider the discretisation er-

ror and computational expense of the numerical method that is used. For the Euler-Maruyama method, these properties are well-known [22]: under standard assumptions (Lipschitz coefficients with linear growth), the discretisation of the Euler method is of order $\Delta t^{1/2}$.

The equation (5.10), on the other hand, is an *ordinary differential equation*. Thus, one can use any convenient out-of-the-box ODE solver to compute X_t. We used the standard ODE solver in MATLAB, which is an adaptive fourth-order Runge-Kutta method. Runge-Kutta methods are known to have a global error of order Δt^4 [81]. Thus, it is likely that the main source of bias in (5.10) comes from truncation of the Brownian series expansion rather than discretisation error.

For a general stochastic differential equation, the Euler-Maruyama method must perform three tasks: evaluate the drift coefficient, evaluate the diffusion coefficient, and generate a draw from a d-dimensional standard normal distribution. The Euler-Maruyama method uses a linear combination of these three quantities to generate the next step in the sample path.

One very simple version of the series expansion method would be to solve the ODE (5.10) using an Euler method. If we truncate the Brownian series expansion after N terms, the Euler method would proceed as follows: First, one has the intital overhead of generating N i.i.d standard normal random variates. Then, at each step, one must evaulate the drift coefficient, the diffusion coefficient, and each of the N basis functions $\{\phi_i\}_{i \leq N}$. A linear combination of these quantities gives us the next value of the process in our solver. We assume the time it takes to generate the linear combination is small in relation to the amount of time it takes to evaluate the functions.

If one wanted to further improve the efficiency of the series expansion method, one could evaluate the basis functions in advance and store the values in a table, referring to them as needed.

We assume the number of steps is large, so that the initial overhead is negligible. The difference in computational expense between these two methods boils down to whether it is cheaper to evaluste the N basis functions or to generate a d-dimensional standard normal variate. This will depend on N, d, and the choice of basis functions used.

5.1.5 Resonance and failure of the series expansion method

One interesting setting in which the series expansion method can fail is when the SDE exhibits resonance at a certain frequency. Resonance causes small periodic oscillations in a function driving a differential equation to me magnified in the solution of that equation. The series expansion approximation relies on the heuristic that small oscillations in the driving Brownian motion are 'irrelevant'. This is manifestly not the case in a system that exhibits resonance.

Perhaps the simplest example of a resonant system is the two-dimensional ODE

$$\frac{d\mathbf{X}_t^1}{dt} = \mathbf{X}_t^2, \tag{5.35}$$

$$\frac{d\mathbf{X}_t^2}{dt} = -\omega\mathbf{X}_t^1 + F\cos(\omega t). \tag{5.36}$$

Here, the natural frequency of the system is ω, which matches the frequency of the driving function $F\cos(\omega t)$. Resonance in the system causes the amplitude of the solution to grow linearly with time.

Suppose now that we replace the driving function with one-dimensional white noise and consider its behaviour on the interval $[0,1]$. The system becomes a linear stochastic differential equation

$$d\mathbf{X}_t^1 = \mathbf{X}_t^2 dt, \tag{5.37}$$

$$d\mathbf{X}_t^2 = -\omega\mathbf{X}_t^1 dt + d\mathbf{W}_t. \tag{5.38}$$

We can expand the white noise in the Fourier cosine series, so that the system becomes

$$d\mathbf{X}_t^1 = \mathbf{X}_t^2 dt, \tag{5.39}$$

$$d\mathbf{X}_t^2 = -\omega\mathbf{X}_t^1 dt + \sqrt{2}\sum_{i=1}^{\infty}\mathbf{Z}_i\cos\left(\left(i-\frac{1}{2}\right)\pi t\right). \tag{5.40}$$

When $\omega = (k-1/2)\pi$ for some whole number k, the system will exhibit resonance at that frequency. If index N at which we truncate the sum is less than k, the system will fail to resonate and the series expansion will fail catastrophically.

Figure 5.5 demonstrates one way in which the series expansion can fail. We constructed an approximate Brownian sample path, and used the Euler-Maruyama scheme to generate a sample path from the system (5.37). We used the same approximate Brownian sample path to compute the values of $\{Z_i\}$ in (5.3). We set $\omega = (9.5)\pi$,

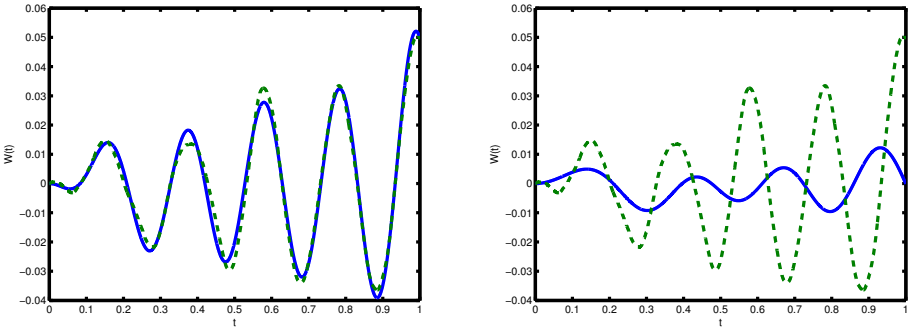


Figure 5.5: Series expansion approximation of a system truncated above the resonant frequency (left) and below the resonant frequency (right). The solid blue line represents the series expansion approximation, while the true process is shown as a broken green line.

matching the frequency of the basis function ϕ_{10}. The leftmost image depicts the resonant behaviour of the approximation when we use $N = 11$ terms in the series expansion, while the rightmost image shows that the approximation fails to resonate when we use $N = 9$ terms.

Chapter 6

MCMC and the series expansion approximation

In Chapter 4, we gave a brief overview of the difficulties that one encounters when attempting to estimate the parameters that govern the evolution of a diffusion process. We mentioned that one major impediment to efficiency in MCMC-based inference algorithms is that it is often difficult to generate sample paths of a diffusion $\mathbf{X}$ that are consistent with observations $\mathbf{Y}_{t_{1:k}}$ of the process. In short, an efficient method of simulating diffusion bridges translates into an efficient way of estimating parameters of a diffusion process.

In this chapter, we describe a novel way of approximating a diffusion bridge. We split the diffusion $\mathbf{X}$ into two components. One component (denoted $\mathbf{X}^{\mathrm{NL}}$, which stands for 'nonlinear component of $\mathbf{X}$') follows the series expansion approximation of Chapter 5.1. The second component (denoted $\mathbf{X}^{\mathrm{L}}$, for 'linear component of $\mathbf{X}$') is a linear process that attempts to correct for the bias introduced by the series expansion approximation. The linearity of this second component allows it to be conditioned to hit a given target.

The strategy we use below is roughly as follows. We draw samples from $\mathbf{X}^{\mathrm{NL}}_{t_k}$ using a Metropolis-Hastings algorithm on the random variables $\{\mathbf{Z}_i\}_{i\leq N}$ in the series expansion approximation of Section 5.1.1. We then condition the linear correction term in such a way that the sum $\mathbf{X}^{\mathrm{NL}}_{t_k}+\mathbf{X}^{\mathrm{L}}_{t_k}$ is consistent with the observation $\mathbf{Y}_{t_k}$. This ensures that the simulated data resemble the observed data, and allows our algorithm to explore the parameter space in an efficient manner.

This chapter is based on material that was published in [18].

6.1 Parametric Diffusion Processes

In this section we develop the basic notation and formalism for the diffusion processes used in this work. First, we assume our data are generated by observing a k-dimensional diffusion process with dynamics

$$d\mathbf{X}_t = \mathbf{a}_\theta(\mathbf{X}_t)dt + \mathbf{B}_\theta d\mathbf{W}_t, \qquad \mathbf{X}_0 \sim p(\mathbf{x}_0), \tag{6.1}$$

where the initial condition is drawn from some known distribution. Observations are assumed to occur at times $t_1, \ldots, t_f$, with $t_k - t_{k-1} := T_k$. We require that $a_\theta : \mathbb{R}^n \to \mathbb{R}^n$ is sufficiently regular to guarantee the existence of a unique strong solution to (6.1), and we assume $\mathbf{B}_\theta \in \mathbb{R}^{n\times d}$. Both terms depend on a set of potentially unknown parameters $\theta \in \mathbb{R}^{d_\theta}$. We impose a prior distribution $p(\theta)$ on the parameters. The driving noise $\mathbf{W}$ is a d-dimensional Brownian motion, and the equation is interpreted in the Itô sense. Observations are subject to independent Gaussian perturbations centered at the true value of $\mathbf{X}$. That is,

$$\mathbf{Y}_{t_k} = \mathbf{X}_{t_k} + \mathbf{V}_{t_k}, \qquad \mathbf{V}_{t_k} \sim \mathcal{N}(0, \mathbf{R}_k) \tag{6.2}$$

As usual, we use the notation $\mathbf{X}$ to refer to the entire sample path of the diffusion, and $\mathbf{X}_t$ to denote the value of the process at time t.

Many systems can be modelled using the form (6.1). Such systems are particularly relevant in physics and natural sciences. In situations where this is not explicitly the case, one can often hope to reduce a diffusion to this form via the Lamperti transform, as we noted in Section 4.1. One can almost always accomplish this in the univariate case, but the multivariate setting is more challenging. Aït-Sahalia [59] characterises the set of multivariate diffusions to which this transform can be applied.

6.2 Related Work

Most approaches to parameter estimation of diffusion processes rely on the Monte-Carlo approximation. Beskos et al. [72] [63] employ a method based on rejection sampling to estimate parameters without introducing any discretisation error. Golightly and Wilkinson [60] extend the work of Chib et al. [82] and Durham and Gallant [67] to construct a Gibbs sampler that can be applied to the parameter estimation problem.

Roughly speaking, Gibbs samplers that exist in the literature alternate between drawing samples from some representation of the diffusion process $\mathbf{X}$ conditional on

parameters θ, and samples from θ conditional on the current sample path of $\mathbf{X}$. Note that draws from $\mathbf{X}$ must be consistent with the observations $\mathbf{Y}_{t_{1:f}}$.

The usual approach to the consistency issue is to make a proposal by conditioning a related diffusion to hit some neighbourhood of the observation $\mathbf{Y}_{t_k}$, then to make a correction via a rejection sampling [62] or a Metropolis-Hastings [82] step. For example, Golightly and Wilkinson [60] sample from a discretised version of the diffusion

$$d\hat{\mathbf{X}}_t = \mathbf{a}_\theta(\hat{\mathbf{X}}_t)dt + \mathbf{B}_\theta\left(\mathbf{B}_\theta(t_k - t) + \mathbf{R}_k\right)^{-1}\left(\mathbf{Y}_{t_k} - \hat{\mathbf{X}}_t - \mathbf{a}(\hat{\mathbf{X}}_t)(t_k - t)\right)dt + \mathbf{B}_\theta d\mathbf{W}_t. \tag{6.3}$$

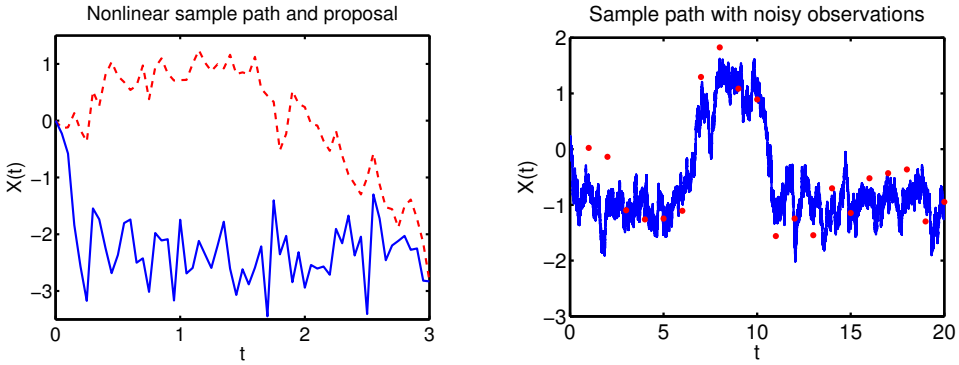
The second term on the right-hand side guarantees that $\hat{\mathbf{X}}_{t_k}$ is approximately normally distributed about $\mathbf{Y}_{t_k}$ with variance $\mathbf{R}_k$. Note, however, that for small values of $\mathbf{R}_k$, the proposal acts like a Brownian bridge. Recall that the Brownian bridge has dynamics

$$d\hat{\mathbf{X}}_t = \frac{1}{(t_k - t)}\left(\mathbf{Y}_{t_k} - \hat{\mathbf{X}}_t\right)dt + \mathbf{B}_\theta d\mathbf{W}_t \qquad t < t_k. \tag{6.4}$$

Thus the behaviour of the sample path of $\hat{\mathbf{X}}$ as a whole may be significantly different from that of $\mathbf{X}$. As the inter-observation time grows, this difference usually gets more pronounced, and the rate of rejection grows accordingly. Figure 6.1 shows the disparity between a sample from a nonlinear process and a sample from the proposal distribution of Durham and Gallant [67], which is essentially a draw from a Brownian bridge (6.4) with a non-standard discretisation. One can see that the target sample path is constrained to stay near the mode $\gamma = 2.5$, whereas the proposal can move more freely. One should expect to make many proposals before finding one that 'behaves' like a typical draw from the true process.

For low-dimensional inference problems, algorithms that employ sequential Monte-Carlo (SMC) methods [83] [65] typically yield good results. However, unlike the Gibbs samplers mentioned above, SMC-based methods often do not scale well with dimension. The number of particles that one needs to maintain a given accuracy is known to scale exponentially with the dimension of the problem [40].

Aït-Sahalia [59, 70] uses a deterministic technique based on Edgeworth expansions to approximate the transition density. Other approaches include variational methods [68, 24] that can compute continuous time Gaussian process approximations to more general stochastic differential systems, as well as various non-linear Kalman filtering and smoothing based approximations [33, 32, 84] .



(a) (b)

Figure 6.1: (a) Sample path of a double well process (see equation (6.16)) with $\alpha = 2,\ \gamma = 2.5,\ B = 2$ (blue line). In a low-noise setting, current Gibbs samplers use proposals that have very similar behaviour to a standard Brownian bridge (dashed red line). These proposals inculde a rejection step, which makes it possible to generate conditioned nonlinear paths. In this case, the behaviour of the proposal is very different to that of the target, and the rate of rejection is high.
(b) Sample path of a double well process (solid blue line) with noisy observations (red dots). We use this as an initial dataset on which to test our algorithm. Parameters are $\alpha = 2, \gamma = 1, B = 1$. Observation errors have variance $\mathbf{R} = .25$.

6.3 MCMC and the series expansion method

We now introduce a method of approximating a nonlinear diffusion that allows us to gain a considerable amount of control over the behaviour of the process. Similar methods have been used for stratified sampling of diffusion processes [85] and for the study of stochastic partial differential equations [73]. One of the major challenges of using MCMC methods for parameter estimation in the present context is that it is typically very difficult to draw samples from a diffusion process conditional on observed data. If one only knows the initial condition of a diffusion, then it is straightforward to simulate a sample path of the process. However, simulating a sample path conditional on both initial and *final* conditions is a challenging problem, as we saw in 2.2.6.

Our approximation separates the diffusion process $\mathbf{X}$ into the sum of a linear and nonlinear component. The linear component of the sum allows us to condition the approximation to fit observed data more easily than in conventional methods. On the other hand, the nonlinear component captures the 'gross' variation of a typical sample path. In this section, we fix a generic time interval $[0,T]$, though one can apply the

same derivation for any given interval $[t_{i-1}, t_i]$ given the initial distribution of $\mathbf{X}$ at time t_{i-1}.

The series expansion approximation introduced in Chapter 5.1 gives us an alternative to the Euler-Maruyama discretisation for sampling approximately from the time-t marginal distribution of a diffusion process. We draw coefficients $\mathbf{Z}_{1:N}$ from a standard normal distribution, and solve the appropriate vector-valued ordinary differential equation (5.10). While the Euler discretisation is the de facto standard method for numerical approximation of SDE, other methods do exist. Kloeden and Platen [22] discuss higher order methods such as the stochastic Runge-Kutta scheme [86].

In the Euler-Maruyama approximation, one discretises the driving Brownian motion into increments $W_{t_i} - W_{t_{i-1}}$. These increments are independent of one another, and can be thought of as Gaussian 'input variables' in their own right. One must typically employ a fine discretisation (i.e. a large number of input variables) to get a good approximation to the true diffusion process. Empirically, we find that one needs far fewer Gaussian inputs Z_i for an accurate representation of $\mathbf{X}_T$ using the series expansion approximation. In one sense, the series expansion approximation reduces the dimensionality of the driving Brownian motion. This can be advantageous: for example, Corlay and Pages [85] employ related ideas to conduct stratified sampling of a diffusion process.

The coefficients $\mathbf{Z}_i$ are also more amenable to interpretation than the Gaussian increments in the Euler-Maruyama expansion. Suppose we have a one-dimensional process in which we use the Fourier cosine basis

$$\phi_k(t) = \sqrt{2/T}\cos((2k-1)\pi t/2T). \tag{6.5}$$

If we change Z_1 while holding the other coefficients fixed, we will typically see a change in the large-scale behaviour of the path. On the other hand, a change in Z_N will typically result in a change to the small-scale oscillations in the path. The separation of behaviours across coefficients gives us a means to obtain fine-grained control over the behaviour of a diffusion process within a Metropolis-Hastings algorithm.

We can improve our approximation by attempting to correct for the fact that we truncated the sum in equation (5.6). Instead of simply discarding the terms $\mathbf{Z}_i\Phi_i$ for $i > N$, we attempt to account for their effect as follows. We assume the existence of some 'correction' process $\mathbf{X}^{\mathrm{C}}$ such that $\mathbf{X} = \mathbf{X}^{\mathrm{NL}} + \mathbf{X}^{\mathrm{C}}$. We know that the dynamics of $\mathbf{X}$ satisfy

$$d\mathbf{X}_t = \mathbf{a}_\theta\left(\mathbf{X}_t^{\mathrm{NL}} + \mathbf{X}_t^{\mathrm{C}}\right)dt + \mathbf{B}_\theta d\mathbf{W}_t. \tag{6.6}$$

Taylor expanding the drift term around $\mathbf{X}^{\mathrm{NL}}$, we see that to first order,

$$\begin{aligned}
d\mathbf{X}_t &\approx \left(\mathbf{a}_\theta\left(\mathbf{X}_t^{\mathrm{NL}}\right) + \mathbf{J}_{\mathbf{a}}(\mathbf{X}_t^{\mathrm{NL}})\mathbf{X}_t^{\mathrm{C}}\right)dt + \mathbf{B}_\theta d\mathbf{W}_t \\
&= \left(\mathbf{a}_\theta\left(\mathbf{X}_t^{\mathrm{NL}}\right) + \mathbf{J}_{\mathbf{a}}(\mathbf{X}_t^{\mathrm{NL}})\mathbf{X}_t^{\mathrm{C}}\right)dt + \mathbf{B}_\theta\left(\sum_{i=1}^{\infty}\mathbf{Z}_i\phi_i(t)\right)dt \\
&= \left(\mathbf{a}_\theta\left(\mathbf{X}_t^{\mathrm{NL}}\right) + \mathbf{B}_\theta\left(\sum_{i=1}^{N}\mathbf{Z}_i\phi_i(t)\right)\right)dt + \mathbf{J}_{\mathbf{a}}(\mathbf{X}_t^{\mathrm{NL}})\mathbf{X}_t^{\mathrm{C}}dt + \mathbf{B}_\theta\left(\sum_{i=N+1}^{\infty}\mathbf{Z}_i\phi_i(t)\right)dt \\
&= d\mathbf{X}_t^{\mathrm{NL}} + \mathbf{J}_{\mathbf{a}}(\mathbf{X}_t^{\mathrm{NL}})\mathbf{X}_t^{\mathrm{C}}dt + +\mathbf{B}_\theta\left(\sum_{i=N+1}^{\infty}\mathbf{Z}_i\phi_i(t)\right)dt.
\end{aligned} \tag{6.7}$$

Here, $\mathbf{J}_{\mathbf{a}}(x)$ is the Jacobian matrix of the function a evaluated at x. This motivates the use of a linear time-dependent approximation to the correction process. We will refer to this linear approximation as $\mathbf{X}^{\mathrm{L}} \approx \mathbf{X}^{\mathrm{C}}$. From (6.7), we see that the dynamics of $\mathbf{X}^{\mathrm{L}}$ satisfy

$$d\mathbf{X}_t^{\mathrm{L}} = \mathbf{J}_{\mathbf{a}}(\mathbf{X}_t^{\mathrm{NL}})\mathbf{X}_t^{\mathrm{L}}dt + \mathbf{B}_\theta\left(\sum_{i=N+1}^{\infty}\mathbf{Z}_i\phi_i(t)\right)dt, \qquad \mathbf{X}_0^{\mathrm{L}} = 0, \tag{6.8}$$

where the driving noise is the 'residual' term

$$\mathbf{R}_t = \mathbf{W}_t - \sum_{i=1}^{N}\mathbf{Z}_i\int_0^t \phi_i(u)du. \tag{6.9}$$

Conditional on $\mathbf{X}^{\mathrm{NL}}$, $\mathbf{X}^{\mathrm{L}}$ is a linear Gaussian process, and equation (6.8) can be solved in semi-closed form. First, we compute a numerical approximation to the solution of the homogenous matrix-valued equation

$$\frac{d}{dt}\Psi(t) = \mathbf{J}_{\mathbf{a}}(\mathbf{X}_t^{\mathrm{NL}})\Psi(t), \qquad \Psi(0) = \mathbf{I}_n. \tag{6.10}$$

One can compute $\Psi^{-1}(t)$ in a similar fashion via the relationship $d\Psi^{-1}/dt = -\Psi^{-1}(d\Psi/dt)\Psi^{-1}$.

We then have

$$\begin{aligned}
\mathbf{X}_t^{\mathrm{L}} &= \Psi(t)\int_0^t \Psi(u)^{-1}\mathbf{B}d\mathbf{R}_u \\
&= \Psi(t)\int_0^t \Psi(u)^{-1}\mathbf{B}d\mathbf{W}_u - \sum_{i=1}^{N}\Psi(t)\left(\int_0^t \Psi(u)^{-1}\mathbf{B}\phi_i(u)du\right)\mathbf{Z}_i.
\end{aligned} \tag{6.11}$$

It follows that $\mathbf{X}^{\mathrm{L}}$ has mean 0 and covariance

$$\begin{aligned}
\Sigma(s,t) = \Psi(s)&\left(\int_0^{s\wedge t}\Psi(u)^{-1}\mathbf{B}\mathbf{B}^{\intercal}\Psi^{\intercal}(u)^{-1}du\right)\Psi^{\intercal}(t) \\
&- \sum_{i=1}^{N}\Psi(s)\left(\int_0^s \Psi(u)^{-1}\mathbf{B}\phi_i(u)du\right)\left(\int_0^t \Psi(u)^{-1}\mathbf{B}\phi_i(u)du\right)^{\intercal}\Psi^{\intercal}(t).
\end{aligned} \tag{6.12}$$

These integrals will rarely be implementable in closed form, and must be approximated numerically – for example, using the trapeziodal approximation method.

The process $\mathbf{X}^{\mathrm{NL}}$ is designed to capture the most significant nonlinear features of the original diffusion $\mathbf{X}$, while the linear process $\mathbf{X}^{\mathrm{L}}$ corrects for the truncation of the sum (5.6), and can be understood using tools from the theory of Gaussian processes. One can think of the linear term as the result of a 'small-noise' expansion *about the nonlinear trajectory*. Small-noise techniques have been applied to diffusions in the past [21], but the method described above has the advantage of being inherently nonlinear.

6.4 Parameter Estimation

In this section, we describe a novel modification of the Gibbs sampler that does not suffer the drawbacks of the linear proposal strategy. In Section 6.5, we demonstrate that for highly nonlinear problems it will perform significantly better than standard methods because of the nonlinear component of our approximation. Our algorithm has the useful property that parameters governing the diffusion coefficient are treated like any other parameters. For many inference algorithms, one must apply the Lamperti transform to 'move' the diffusion parameters into the drift function. In contrast, our algorithm can be applied to a diffusion process in its given form.

Suppose for now that we make a single noiseless observation at time $t_1 = T$ (for ease of notation, we will assume that observations are uniformly spaced through time with $t_{i+1} - t_i = T$, though this is not necessary). Our aim is to sample from the posterior distribution

$$p\left(\theta, \mathbf{Z}_{1:N} | \, \mathbf{X}^{\mathrm{NL}}_{t_1} + \mathbf{X}^{\mathrm{L}}_{t_1} = \mathbf{Y}_{t_1}\right) \propto \mathcal{N}(\mathbf{Y}_{t_1} \mid \mathbf{X}^{\mathrm{NL}}_{t_1}, \Sigma(t_1, t_1)) \mathcal{N}(\mathbf{Z}_{1:N}) p(\theta). \tag{6.13}$$

We adopt the convention that $\mathcal{N}(\cdot | \, \mu, \, \Sigma)$ represents the normal distribution with mean μ and covariance Σ, whereas $\mathcal{N}(\cdot)$ represents the standard normal distribution. Note that we have left dependence of Σ_1 on the path of $\mathbf{X}^{\mathrm{NL}}$ implicit. The right-hand side of this expression allows us to evaluate the posterior up to proportionality; hence it can be targeted with a Metropolis-Hastings sampler.

With multiple observations, the situation is similar. However, we now have a set of Gaussian inputs $\mathbf{Z}^{(k)}_{1:N}$ for each transition $\hat{\mathbf{X}}_k | \hat{\mathbf{X}}_{k-1}$. If we attempt to update θ and $\{\mathbf{Z}^{(k)}_{1:N}\}_{k \leq f}$ all at once, the rate of rejection will be unacceptably high. For this reason, we update each $\mathbf{Z}^{(k)}_{1:N}$ in turn, holding θ and the other Gaussian inputs fixed. We draw $\mathbf{Z}^{(k)*}_{1:N}$ from the proposal distribution, and compute $\mathbf{X}^{\mathrm{NL}*}_k$ with initial condition $Y_{t_{k-1}}$.

We also compute the covariance $\Sigma_k^*(t_k,t_k)$ of the linear correction given the path of $\mathbf{X}^{\mathrm{NL}}$ from time t_{k-1} to time t_k. The acceptance probability for the update is

$$\alpha = 1 \wedge \frac{\mathcal{N}(\mathbf{Y}_{t_k} \mid \mathbf{X}_{t_k}^{\mathrm{NL}*}, \Sigma_k^*(t_k,t_k))\mathcal{N}(\mathbf{Z}_{1:N}^{(k)*})p(\mathbf{Z}_{1:N}^{(k)*} \to \mathbf{Z}_{1:N}^{(k)})}{\mathcal{N}(\mathbf{Y}_{t_k} \mid \mathbf{X}_{t_k}^{\mathrm{NL}}, \Sigma_k(t_k,t_k))\mathcal{N}(\mathbf{Z}_{1:N}^{(k)})p(\mathbf{Z}_{1:N}^{(k)} \to \mathbf{Z}_{1:N}^{(k)*})} \tag{6.14}$$

After updating the Gaussian inputs, we make a global update over all observations for the θ parameter. The acceptance probability for this move is

$$\alpha = 1 \wedge \prod_{k=1}^{f} \frac{\mathcal{N}(\mathbf{Y}_{t_k} \mid \mathbf{X}_{t_k}^{\mathrm{NL}*}, \Sigma_k^*(t_k,t_k))p(\theta^*)p(\theta^* \to \theta)}{\mathcal{N}(\mathbf{Y}_{t_k} \mid \mathbf{X}_{t_k}^{\mathrm{NL}}, \Sigma_k(t_k,t_k))p(\theta)p(\theta \to \theta^*)}, \tag{6.15}$$

where $\mathbf{X}_{t_k}^{\mathrm{NL}*}$ and $\Sigma_k^*(t_k,t_k)$ are computed using the proposed value of θ^*.

We noted earlier that when j is large, $\mathbf{Z}_j$ governs the small-time oscillations of the diffusion process. One should not expect to gain much information about the value of $\mathbf{Z}_j$ when we have large inter-observation times. We find this to be the case in our experiments - the posterior distribution of $\mathbf{Z}_{j:N}$ approaches a spherical Gaussian distribution when $j > 3$. For this reason, we employ a Gaussian random walk proposal in $\mathbf{Z}_1$ with stepsize $\sigma_{\mathrm{RW}} = .45$, and proposals for $\mathbf{Z}_{2:N}$ are drawn independently from the standard normal distribution.

In the presence of observation noise, we proceed roughly as before. Recall that we make observations $\mathbf{Y}_{t_k} = \mathbf{X}_{t_k} + \mathbf{V}_{t_k}$. We draw proposals $\mathbf{Z}_{1:N}^{(k)*}$ and $\mathbf{V}_k^*$. The initial condition for $\mathbf{X}_k^{\mathrm{NL}}$ is now $\mathbf{Y}_{t_{k-1}} - \mathbf{V}_{t_{k-1}}$. However, one must make an important modification to the algorithm. Suppose we propose an update of $\hat{\mathbf{X}}_{t_k}$ and it is accepted. If we subsequently propose an update for $\hat{\mathbf{X}}_{t_{k+1}}$ and it is rejected, then the initial condition for $\hat{\mathbf{X}}_{t_{k+1}}$ will be inconsistent with the current state of the chain (it will be $\mathbf{Y}_{t_k} - \mathbf{V}_{t_k}$ instead of $\mathbf{Y}_{t_k} - \mathbf{V}_{t_k}^*$). For this reason, we must propose joint updates for $(\hat{\mathbf{X}}_{t_k}, \mathbf{V}_{t_k}, \hat{\mathbf{X}}_{t_{k+1}})$. If the variance of the observation noise is high, it may be more efficient to target the joint posterior distribution $p\left(\theta, \{\mathbf{Z}_{1:N}^k, \mathbf{X}_{t_k}^{\mathrm{L}}\} \mid \mathbf{Y}_{1:f}\right)$.

6.5 Numerical Experiments

The double-well diffusion is a widely-used benchmark for nonlinear inference problems [24, 69, 78, 79]. It has been used to model systems that exhibit switching behaviour or bistability [21, 87]. It possesses nonlinear features that are sufficient to demonstrate the shortcomings of some existing inference methods, and how our approach overcomes these issues. The dynamics of the process are given by

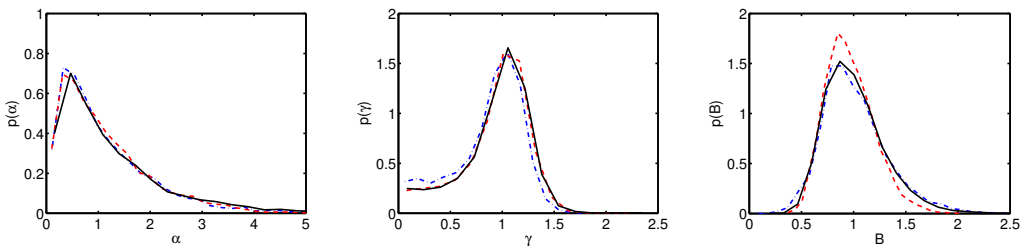
$$dX_t = \alpha X_t \left(\gamma^2 - X_t^2\right) dt + B dW_t. \tag{6.16}$$

The process X has a bimodal stationary distribution, with modes at $x = \pm\gamma$. The parameter α governs the rate at which sample trajectories are 'pushed' toward either mode. If B is small in comparison to α, mode-switching occurs relatively rarely.

Figure 6.1(b) shows a trajectory of a double-well diffusion over 20 units of time, with observations at times $\{1, 2, \ldots, 20\}$. We used the parameters $\alpha = 2$, $\gamma = 1$, $B = 1$. The variance of the observation noise was set to $\mathbf{R} = .25$.

As we mentioned earlier, particle MCMC performs well in low-dimensional inference problems. For this reason, the results of a particle MCMC inference algorithm (with $N = 1,000$) particles are used as 'ground truth'. Our algorithm used $N = 3$ Gaussian inputs with a linear correction. We used the Fourier cosine series (6.5) as an orthonormal basis. We compare our Gibbs sampler to that of Golightly and Wilkinson [60], for which we use an Euler discretisation with stepsize $\Delta t = .05$. Each algorithm drew $70,000$ samples from the posterior distribution, moving through the parameter space in a Gaussian random walk. We used exponential priors for γ, α and B. The exponential density function is proportional to $\exp(-\lambda)$. We set $\lambda = 1$ for α and B, and $\lambda = 4$ for the parameter γ.

For this particular choice of parameters, both Gibbs samplers give a good approximation to the true posterior. Figure 6.2 shows empirical density plots of the marginal posterior distributions of (α, γ, B) for each algorithm.



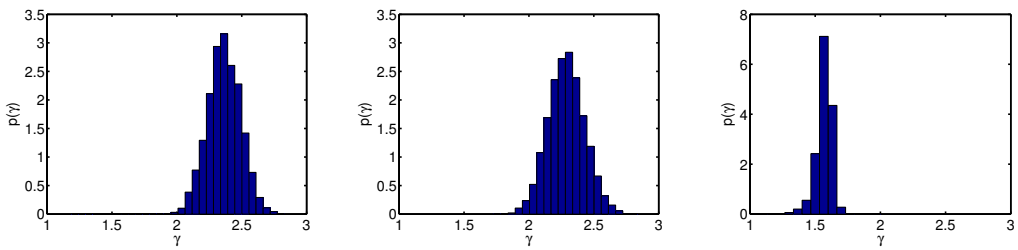
(a) $p(\alpha|Y_{1:20})$ (b) $p(\gamma|Y_{1:20})$ (c) $p(B|Y_{1:20})$

Figure 6.2: Marginal posterior distributions for (α, γ, B) conditional on observed data. The solid black line is the output of a particle MCMC method, taken as ground truth. The broken red line is the output of the linear proposal method, and the broken and dotted blue line is the density estimate from the coloured noise expansion method. We see that both methods give a good approximation to the ground truth.

Gibbs samplers that have been used in the past rely on making proposals by conditioning a linear diffusion to hit a target, and subsequently accepting or rejecting those proposals. Over short timescales, or for problems that are not highly nonlinear, this can

be an effective strategy. However, as the timescale increases, the proposal and target become quite dissimilar (see Figure 6.1(a)).

To investigate how the difficulty of inference increases as the inter-observation time grows, we simulate a double well process with $(\alpha, \gamma, B) = (2, 2.5, 2)$. We make noisy observations with $t_k - t_{k-1} = 3$ and $\mathbf{R} = .1$. The algorithms target the posterior distribution over γ, with α and B fixed at their true values. From our previous discussion, one might expect the linear proposal strategy to perform poorly in this more nonlinear setting. This is indeed the case. As in the previous experiment, we used a linear proposal Gibbs sampler with Euler stepsize $dt = 0.05$. In the 'path update' stage, fewer than $.01\%$ of proposals were accepted. On the other hand, the series expansion method used $N = 7$ Gaussian inputs with a linear correction and was able to approximate the posterior distribution accurately. Figure 6.3 shows empirical density plots of the results. Note the different vertical scaling of the rightmost plot.



(a) PMCMC estimate (b) Series expansion estimate (c) Standard proposal estimate

Figure 6.3: $p(\gamma|Y_{1:10}, B, \alpha)$ after ten observations with a relatively large inter-observation time ($T = 3$). We drew data from a double well process with $(\alpha, \gamma, B) = (2, 2.5, 2)$. The series expansion method matches the ground truth, whereas the linear proposal method is inconsistent with the data.

The series expansion method has the drawback that, in some sense, one is using the 'wrong' model for inference. By expanding the white noise driving the process and truncating it, one introduces a bias that is difficult to quantify. Of course, one could make the same argument for any method of descritising a diffusion process. The crucial difference is that asymptotic results are available for standard methods.

On the other hand, one can use powerful ODE solvers (e.g. high order adaptive Runge-Kutta schemes) for the 'smoothed' process driven by the truncated white noise. These solvers are not usually available for general diffusion processes, and one must often rely on stochastic analogues of the Euler scheme.

It is difficult to provide a general analysis of the runtime of the algorithms since there are many variables that must be accounted for. Among them are the number of particles and effective sample size of the particle MCMC algorithm, the rejection rate of the Metropolis step in the MCMC algorithms, and cost of generating a single diffusion sample path (which varies according to the discretisation used, or to the order of the series expansion approximation).

In any case, we expect the series expansion approximation to run more slowly than Golightly and Wilkinson's MCMC scheme. The reason for this is that one must solve the ODE (6.10) for each sample path generated in the MCMC sampler. This is an ODE in $n \times n$ dimensions, and may be costly to solve in high-dimensional problems.

As a further demonstration of the series expansion MCMC algorithm, we consider the *stochastic Hopf bifurcation*. This is a two-dimensional diffusion whose dynamics satisfy

$$dX_{1,t} = \Big(aX_{1,t} - X_{2,t}\Big)dt - X_{1,t}\Big(X_{1,t}^2 + X_{2,t}^2\Big)dt + bdW_{1,t} \tag{6.17}$$

$$dX_{2,t} = \Big(aX_{2,t} + X_{1,t}\Big)dt - X_{2,t}\Big(X_{1,t}^2 + X_{2,t}^2\Big)dt + bdW_{2,t} \tag{6.18}$$

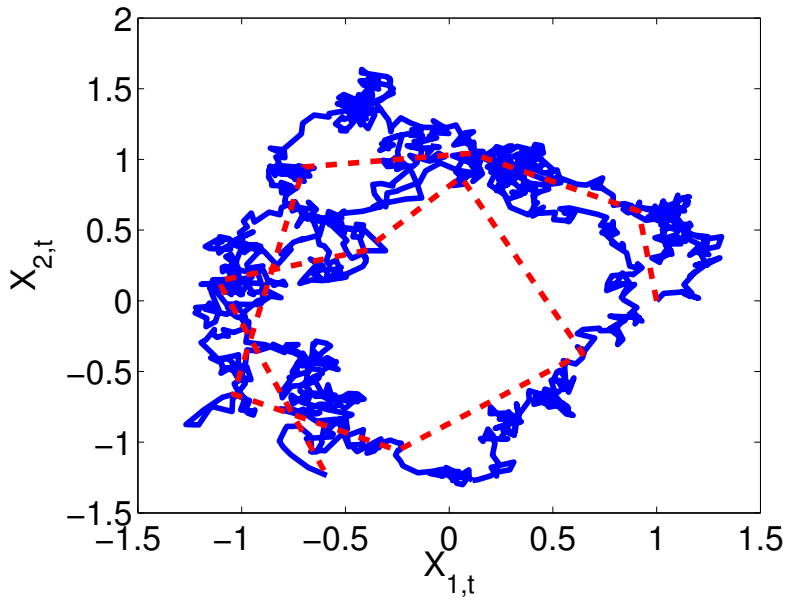
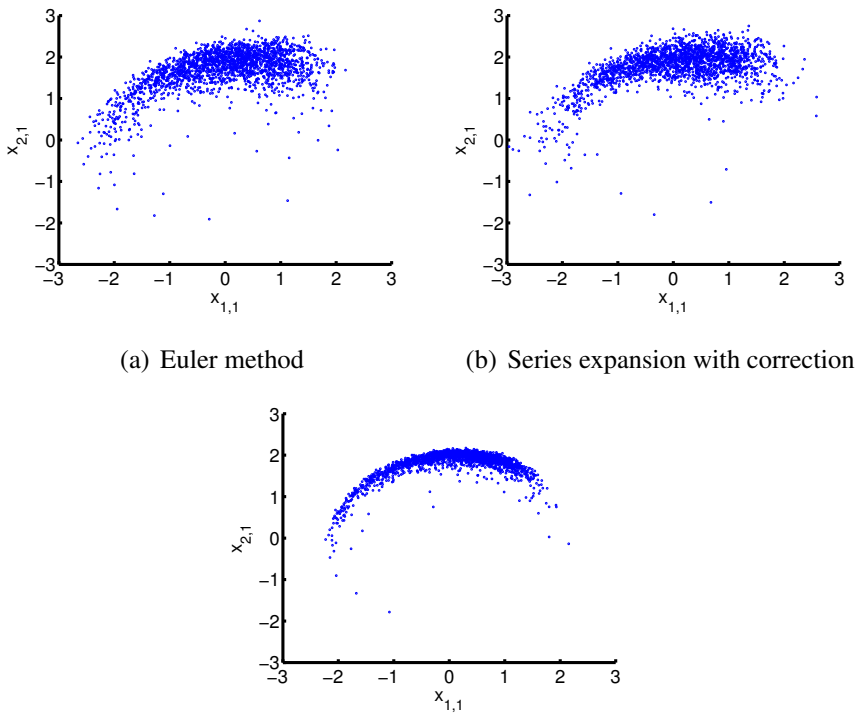


Figure 6.4: Sample path of the stochastic Hopf bifurcation with $a = 1$, $b = .5$ (blue), together with ten observations (broken red line).

Here, $a \in \mathbb{R}$, $b > 0$, and $W_{1,t}$ and $W_{2,t}$ are independent Brownian motions. For positive values of a, sample paths of this process rotate around the circle $X_{1,t}^2 + X_{2,t}^2 =$

$\sqrt{a}$. The stochastic Hopf bifurcation is a simple example of an oscillatory system. It has been used in climate modelling [5] and simple models of cardiac rhythm [88] among other applications.

We truncated the series expansion method at $N = 2$ terms for both Brownian motions. This may seem like a crude approximation, but along with the correction term, it is sufficient for our purposes. Figure 6.5 shows scatter plots of an Euler-Maruyama approximation and a series expansion approximation with correction. We used $a = 4$, $b = .5$ and $T = 1$. Empirically, one can see that the distributions are close to one another. For the parameters used in the parameter estimation experiment below, the approximation is also effective. However, the 'banana' shape observed with $a = 4$ gives a striking demonstration of the effectiveness of the correction term.



(a) Euler method

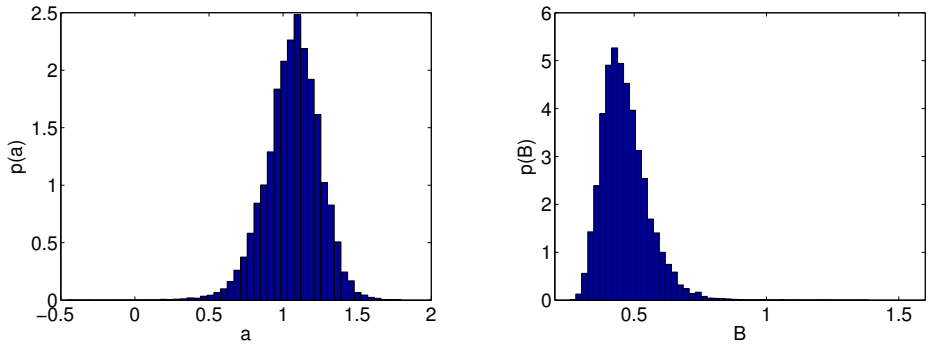
(b) Series expansion with correction

(c) Series expansion, no correction

Figure 6.5: Scatter plot of 2,000 draws from the stochastic Hopf bifurcation at time $T = 1$. The Euler method is shown on the left. The series expansion method (truncated at $N = 2$) with correction is shown on in the center. The series expansion method ($N = 2$) with no correction is shown on the right. The first two distributions coincide closely, whereas the series expansion method with no correction is very different. This shows the effectiveness of the correction term.

For our parameter estimation experiment, we set $a = 1$ and $b = .5$, and generated a sample path from the stochastic Hopf bifurcation. We made ten *exact* observations spaced $T = 1$ unit of time apart (see Figure 6.4). In one sense, the 'exact observations' inference problem is more difficult than the 'noisy oservations problem'. This is because of the difficulty of simulating sample paths that are consistent with the data (see Section 2.2.6 for a discussion on conditioned diffusion processes). The standard bootstrap filter, for example, is not applicable in this setting.

We used a random walk proposal in a, b and Z_1, and sampled Z_2 from the prior. Figure 6.6 shows 100,000 samples from the posterior distribution as computed by the series expansion MCMC method. We see that the distributions are centered around the true values of the data.



(a) Posterior for parameter a (b) Posterior for parameter b

Figure 6.6: Empirical density plots of 100,000 samples from the marginal posterior distributions for parameters a and b in the Hopf bifurcation experiment.

6.6 Discussion and Future Work

We have seen that the standard linear proposal/correction strategy can fail for highly nonlinear problems. Our inference method avoids the linear correction step, instead targeting the posterior over input variables *directly*. With regard to computational efficiency, it is difficult to give an authoritative analysis because both our method and the linear proposal method are complex, with several parameters to tune. In our experiments, the algorithms terminated in a roughly similar length of time (though no serious attempt was made to optimise the runtime of either method).

With regard to our method, several questions remain open. The accuracy of our

algorithm depends on the choice of basis functions $\{\phi_i\}$. At present, it is not clear how to make this choice optimally in the general setting. In the linear case, it is possible to show that one can achieve the accuracy of the Karhunen-Loeve decomposition, which is theoretically optimal. As we saw in Chapter 5.1, one can also set the error at a single time t to zero with a judicious choice of a single basis function in the univariate case.

We used a Taylor expansion to compute the covariance of the correction term. However, it may be fruitful to use more sophisticated ideas, collectively known as statistical linearisation methods. In this chapter, we restricted our attention to processes with a state-independent diffusion coefficient so that the covariance of the correction term could be computed. We may be able to extend this methodology to process with state-dependent noise - certainly one could achieve this by taking a 0-th order Taylor expansion about $\mathbf{X}^{\mathrm{NL}}$. Whether it is possible to improve upon this idea is a matter for further investigation.

Chapter 7

The series expansion unscented Kalman filter

In Chapter 3, we discussed the *filtering problem*, in which one makes noisy observations $\{\mathbf{Y}_{t_k} \in \mathbb{R}^s\}_{k\geq 1}$ of a process and is faced with the task of computing the expectation $\mathbb{E}[\phi(\mathbf{X}_t)|\mathbf{Y}_{t_1},\ldots,\mathbf{Y}_{t_f}]$ for a given function ϕ when $t > t_f$.

In mathematical terms, the model for measurements of this type can often be written as

$$\mathbf{Y}_{t_k} = \mathbf{h}(\mathbf{X}_{t_k}) + \mathbf{V}_{t_k}, \tag{7.1}$$

for some known 'observation function' $\mathbf{h}$ with Gaussian measurement noise $\mathbf{V}_{t_k} \sim \mathcal{N}(0, \mathbf{R}_k)$.

For simplicity, we assume that the distribution of $\mathbf{X}_t$ conditional on the observations has a density with respect to Lebesgue measure. For filtering problems where this is not the case, such as when part of the system is observed without error, much of our analysis can be applied with only minor modifications. The estimation problem can be solved for arbitrary ϕ provided that we can compute the filtering density $p_{\mathbf{X}_t}(x\,|\{\mathbf{Y}_{t_k} : t_k \leq t\})$ for all t.

It is only in a small number of special cases that the filtering density can be described using a finite number of parameters. When the SDE is linear and the function $\mathbf{h}$ in the measurement model is linear, then the Kalman filter can be used to compute the exact solution [25]. Certain other filtering problems also admit closed-form solutions (see, for example, the Beneš filter [29]). However, closed-form filters are rare, and in most cases one must approximate the filtering distribution in some manner. For example, one can discretise the signal and employ a particle filter [89, 44, 90, 91], which uses Monte Carlo samples to approximate the filtering distribution. Other ap-

proaches include variational filtering [92], homotopy filtering [93], and path integral filtering [94].

Another general technique is to take a parametric set of tractable densities (for example a set of densities within the exponential family) and find the density within that set that most closely matches the filtering density. This approach, introduced in [95], is known as *assumed density filtering*.

In this chapter, we will attempt to compute statistics of the Gaussian distribution that most closely matches the filtering distribution. This particular special case of assumed density filtering is known as *Gaussian filtering* [96]. There are a number of ways to approach the problem. The *extended Kalman filter* (EKF) [32] uses a Taylor series approximation to the nonlinearities in SDE and measurement model (see Section 3.1.2). The *unscented Kalman filter* (UKF) described in Section 3.1.1 uses a set of sigma-points for computing the mean and covariance of the Gaussian approximation [41, 97, 33]. Quadrature and cubature based filters [96, 98, 80, 34] use Gaussian numerical integration for computing the mean and covariance. The Gaussian assumption is a natural one when the filtering distribution is known to be unimodal. However, it may lead to significant errors for certain multimodal distributions. It is not advisable to apply a Gaussian filter blindly, without considering the possibility of encountering a multimodal filtering distribution.

The commonly used approaches to filtering in continuous-discrete systems can be divided into two categories: one possibility is that the SDE is first discretised using methods such as Itô–Taylor series or a stochastic Runge–Kutta discretisation [22], [90]. Discrete-time filtering algorithms are then applied to the discretised process. The alternative is that an approximate filter is formed that operates in continuous time, and that filter is discretised. The relative merits of these approaches were recently studied in [99].

In this chapter, we describe a different approach. The series expansion approximation of Chapter 5.1 has the advantage of approximating the transition map of the diffusion $\mathbf{X}$ in finite-dimensional terms. We can view the solution of (5.10) as a function

$$\hat{\mathbf{X}}_t = \mathbf{f}(t, \mathbf{X}_0, \mathbf{Z}_1, \ldots, \mathbf{Z}_N). \tag{7.2}$$

In essence, the time-t distribution of the process $\hat{\mathbf{X}}$ can be interpreted as the image of a $d \times N$-dimensional Gaussian distribution under a nonlinear transform. We can apply sigma point methods to $\mathbf{f}$ to estimate the mean and variance of $\mathbf{X}_t$. The image of each sigma point under the transform $\mathbf{f}(\cdot)$ is computed by solving (5.10).

Our method requires *one* application of the unscented transform per observation (though this is generalised in Section 7.2.2). This is in contrast to the standard UKF, which discretises the system first, then iteratively applies the unscented transform at each timestep.

Gaussian filters that currently exist in the literature typically rely on discretisation of the signal. The time-t distribution of the discretised signal is repeatedly projected onto the set of Gaussian distributions, for example through moment matching or by minimising some form of generalised metric as in [24]. Our methodology avoids repeated projection onto the space of Gaussian random variables during the prediction phase. For this reason we expect our new prediction step to outperform the prediction steps of existing methods when the inference problem is sufficiently nonlinear.

The chapter is structured as follows. In Section 7.1, we describe our method of approximating the time-t marginal distribution of a diffusion process, and we show how the approximation can be exploited to construct a novel Gaussian filter. The accuracy of this approximation is investigated in Section 7.2, and we show that our filter performs well on a high-dimensional nonlinear problem. In Section 7.3, we review our work and discuss some questions that arise as a result of the study.

This chapter is based on material that was published in [19].

7.1 The series expansion filter

Our algorithm proceeds as follows. We assume we have a Gaussian approximation $\mathcal{N}(\mathbf{m}_{t_{k-1}}, \mathbf{P}_{t_{k-1}})$ to the filtering distribution at time t_{k-1}. We wish to compute the filtering distribution at time t. If $t < t_k$, we compute the predictive distribution. If $t = t_k$, we must also update the predictive distribution with the information gained from our observation $\mathbf{Y}_{t_k}$.

We choose a set $\{\sigma_j\}$ of sigma points to represent the joint distribution of the state and the random coefficients $\{\mathbf{Z}_i\}$ in (5.10). Each sigma point can be thought of as a vector of dimension $n + d \times N$,

$$\sigma^j = (\sigma_x^j, \sigma_z^j). \tag{7.3}$$

Here, the first n elements σ_x^j of the vector σ^j are the sigma points for the initial condition for the ODE (5.10), that is, the sigma points that represent $\mathcal{N}(\mathbf{m}_{t_{k-1}}, \mathbf{P}_{t_{k-1}})$. The remaining $d \times N$ elements σ_z^j are the sigma points corresponding to an N-term expansion of a d-dimensional Brownian motion. Together, these data determine an initial value problem. For each sigma point σ^j, we solve the ordinary differential

equation (5.10). The initial condition is $\hat{\mathbf{X}}_{t_{k-1}} = \sigma_x^j$ and the coefficients representing $\{\mathbf{Z}_i\}_{i \leq N}$ are formed from the appropriate subvectors of σ_z^j (each one having length d). At time T, the solution is an n-dimensional vector

$$\hat{\mathbf{X}}_T^j = \hat{\mathbf{X}}(T, \sigma_x^j, \sigma_z^j). \tag{7.4}$$

We treat the solution at time T of the initial value problem as the image of the sigma point σ^j. The set of vectors $\{\hat{\mathbf{X}}_t^j\}$ can be thought of as a discrete approximation to the predictive distribution. We can use these vectors to compute an estimate of $\mathbf{m}_t$ and $\mathbf{P}_t$, though the specific computation depends on the choice of sigma-point method. This methodology is in marked contrast to the sigma point Kalman filters of Section 3.2. These rely on discretisation of the signal dynamics and sigma point approximation of the Brownian increment $\mathbf{W}_{t+\Delta t} - \mathbf{W}_t$ at each timestep, or a limiting case of this discretisation as $\Delta t \to 0$.

We summarise our algorithm in pseudocode as follows:

for $k = 1 : m$ **do**
  Set $\mathbf{m}_\sigma = (\mathbf{m}_{t_{k-1}}, \mathbf{0}_{1\times(Nd)})$
  Set $\mathbf{P}_\sigma = \begin{pmatrix} \mathbf{P}_{t_{k-1}} & \mathbf{0}_{n\times(Nd)} \\ \mathbf{0}_{(Nd)\times n} & I_{(Nd)\times(Nd)} \end{pmatrix}$
  Generate $2(n+Nd)+1$ sigma points, with weighted mean $\mathbf{m}_\sigma$ and weighted covariance $\mathbf{P}_\sigma$
  for Each sigma point $\sigma^{(j)}$ **do**
    Set $x_0 = \sigma_{1:n}^{(j)}$.
    Set $\mathbf{Z}_{1:N} = \sigma_{n+1:n+(Nd)}^{(j)}$ (reshaping the right-hand side into a $d \times N$ matrix if appropriate).
    Solve numerically Equation (5.10). Let $\mathbf{X}_T^{(j)}$ be the value of the solution after T units of time.
    Set $\mathcal{Y}_j = \mathbf{h}(\mathbf{X}_T^{(j)})$.
  end for
  Predict the mean and variance of the incoming observation using (3.38) and (3.39).
  Upon arrival of the observation $\mathbf{Y}_{t_k}$, update the mean $\mathbf{m}_{t_k}$ and variance $\mathbf{P}_{t_k}$ of the filtering distribution using (3.47).
end for

7.2 Numerical experiments

A general analysis of the error induced by the series expansion approximation is difficult. One cannot easily exploit the usual tools from the theory of stochastic processes. In general, the truncated driving noise does not possess the Markov property, nor is it a martingale. The truncated driving noise is, however, a Gaussian process, and this structure is exploited in [100] to demonstrate convergence to the true SDE. In the first part of this section we present a numerical investigation into the approximation error.

We then compare the series expansion UKF with the cubature Kalman filter, which was found to be the most accurate and numerically stable amongst standard unscented transform-based filters in this context. There is already a considerable amount of theoretical and empirical evidence in the literature that sigma point methods outperform the extended Kalman filter, especially in tracking models such as the one described below (see, for example, [97] [80] [31]). In addition, one must compute the gradient of the drift function in order to implement the EKF. For some processes, this can be cumbersome. In contrast, our algorithm can be used as a 'black box' filter. We compare our results with the UKF rather than the EKF to provide the most informative experiments. In these experiments, we use a Stratonovich-to-Itô correction term to modify the dynamics of our approximation, so that the solution coincides with the Itô dynamics [22].

7.2.1 Filtering Experiments

We will test our filtering algorithm on on a model of an aircraft turning in the (x_1, x_3) plane, first seen in Section 5.1.4. Recall that dynamics of the seven-dimensional system are given by

$$d\mathbf{X}_t = \mathbf{a}(\mathbf{X}_t)dt + \mathbf{b}(\mathbf{X}_t)d\mathbf{W}_t, \tag{7.5}$$

with

$$\mathbf{a}(x_{1:7}) = \begin{pmatrix} x_2 \\ -x_7x_4 \\ x_4 \\ x_7x_2 \\ x_6 \\ 0 \\ 0 \end{pmatrix} \tag{7.6}$$

$$\mathbf{b}(x_{1:7}) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ \frac{\sqrt{1+x_2^2}}{v} & \frac{\sqrt{1+x_4^2}}{v_{xy}} & \frac{\sqrt{(1+x_2^2)(1+x_6^2)}}{vv_{xy}} & 0 \\ 0 & 0 & 0 & 0 \\ \frac{\sqrt{1+x_4^2}}{v} & -\frac{\sqrt{1+x_2^2}}{v_{xy}} & \frac{\sqrt{(1+x_4^2)(1+x_6^2)}}{vv_{xy}} & 0 \\ 0 & 0 & 0 & 0 \\ \frac{\sqrt{1+x_6^2}}{v} & 0 & -\frac{v_{xy}}{v} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{7.7}$$

The coordinates $x_1, x_3, and x_5$ are understood to represent the x, y, and z coordinates of the aircraft in Km. The coordinates $x_2, x_4, and x_6$ represent velocity in Km/sec, and the coordinate x_7 represents the turn rate of the aircraft in degrees per second.

As the nonlinearity of the system increases, the speed at which the filtering distribution deviates from Gaussianity should also increase. Intuitively, this means the amount of information that the conventional UKF 'throws away' at each timestep grows with nonlinearity of the system. The series expansion method avoids this issue by targeting the predictive density at a given time directly without any intermediate projection onto the space of Gaussian distributions. As a result, we should expect the series expansion filter to outperform the conventional UKF in systems that are more highly nonlinear.

To test this hypothesis, we set the covariance of the four-dimensional Brownian motion driving the aircraft model to $\mathrm{Cov}(\mathbf{W})(t) = \mathrm{Diag}(10, 0.2, 0.2, Q_W^2)t$. The quantity Q_W determines the variance of the turn rate of the aircraft. We use it as a proxy for the degree of nonlinearity of the system. We chose a number of values for Q_W, ranging between $Q_W = 0.1$ and $Q_W = 1.1$. For each value of the variance, we simulated 1000 trajectories for the aircraft, running both filters on each trajectory. For each trajectory, the initial condition was drawn from a Gaussian distribution with mean $m_0 = (1000, 0, 2650, 150, 200, 0, 6)$. The standard deviation of each component was set to 100, with the exception of the standard deviation of $\mathbf{X}_7(0)$ (recall that this notation denotes the seventh component of the vector at time 0, rather than the value at time 7). This was set to 0.1. All components were assumed to be uncorrelated initially.

For each trajectory, we simulated $n^{\mathrm{obs}} = 20$ observations, spaced $T = 8$ units of time apart. The observation function $\mathbf{h}$ models radar signals arriving at a dish. For this

reason, we assume observations arrive in spherical coordinates, so that $\mathbf{h}$ is given by

$$\mathbf{h}(x_{1:7}) = \begin{pmatrix} \sqrt{x_1^2 + x_3^2 + x_5^2} \\ \tan^{-1}(x_3/x_1) \\ \tan^{-1}(x_5/\sqrt{x_1^2 + x_3^2}). \end{pmatrix} \tag{7.8}$$

The covariance matrix of the observation noise was set to $\mathbf{R} = \mathrm{diag}(50, 0.1, 0.1)$.

For the standard unscented Kalman filter, an Itô-Taylor scheme such as the one proposed in [80] is impractical to implement as a result of the state-dependent noise. This is due to the presence of iterated stochastic integrals in which the integrand is a function of $\mathbf{X}_t$ (see [22]). Even the simplified order 2.0 Itô-Taylor scheme proposed in [22] is cumbersome to implement. For an n-dimensional process, we need to compute $n^2 + 2n + 1$ terms involving derivatives of the coefficient functions (in our case, $n = 7$ so this means 64 terms). The simplified scheme also involves a number of Bernoulli random variables, and it is not immediately clear how one would incorporate these into an unscented filter.

We chose to use the limiting scheme first proposed in [33]. The system of ODEs (3.46) was solved by a fourth order Runge-Kutta scheme. The number of Runge-Kutta steps used did not appear to affect the error appreciably. However, with a large step size the predicted covariance can fail to be positive definite, which causes the filter to diverge. We found that a good compromise between computational cost and the divergence issue was to choose a smaller step-size for more highly nonlinear parameter settings. For this reason, we used $200Q_{\mathbf{W}}$ steps per unit time.

The system of ordinary differential equations (5.10) defining the series expansion method was solved numerically using the Dormand-Prince Runge-Kutta method. This is the default ODE solver implemented in MATLAB. It is an adaptive algorithm, and the number of timesteps used depends on the integrand.

Run-time of either algorithm depends on a number of factors. The main factors that determine computation time are the numerical method used (and the number of timesteps in that method), and the number N of series expansion terms. In our setup, we found that the series expansion method could run anywhere from four times as fast to three times slower than the standard unscented filter. We stress, however, that no effort was made to push either method to the limit of efficiency.

For the standard unscented filter, we set $\alpha = 1$, $\kappa = 0$ and $\beta = 0$: see Section 3.1.2 for the definition of these parameters in the context of the unscented transform. This choice of tuning parameters is also known as the cubature Kalman filter [98, 80].

Various other parameter settings produced similar results, though these settings were most stable and most accurate.

For the series expansion method, we used the orthonormal basis (5.34) with $T = 8$, and used $N = 8$ basis functions for each component of the Brownian motion.

The series expansion filter takes one large step instead of many small ones. As such, one can expect that the target distribution is less like a Gaussian distribution. We found that 'tweaking' the standard parameters slightly improved performance, though not dramatically. We set $\alpha = 1, \kappa = -32, \beta = 0$ so that $\lambda = 7$. Our motivation for this choice is given in Section 7.3.

For any given sample path, we compute the root mean squared error for the position, velocity and turn rate:

$$\varepsilon_c = \sqrt{\frac{1}{n^{\text{obs}} l} \sum_{k=1}^{n^{\text{obs}}} \left(\mathbf{X}_c(t_k) - \mathbf{m}_c(t_k)\right)^\top \left(\mathbf{X}_c(t_k) - \mathbf{m}_c(t_k)\right)}, \tag{7.9}$$

This results in a collection $\{\varepsilon^{(i)}\}_{i \leq n^{\text{obs}}}$ of vectors recording the errors for each sample path. Here, $\mathbf{m}_c(t_k)$ is the mean of the filtering distribution at time t_k. The value of c depends on the error component. For position errors, $c = (1,3,5)$. For velocity errors, $c = (2,4,6)$, and for turn rate errors, $c = 7$. We set $l = 3$ for the position and velocity errors and $l = 1$ for the turn rate error. Mean filter errors and divergences are reported in Table 7.2. A filter was deemed to have diverged if the RMSE position error was greater than 1 km. When this occurred, the corresponding value of $\varepsilon^{(i)}$ was not included in the average.

Both the series expansion filter and unscented filter can diverge and lose track of the signal, in which case the error becomes very large. Even if divergences are discarded, a few large errors can still dominate the average. For this reason, we report the median over all runs of the absolute error for each component in Figure 7.1.

We report quartiles of the empirical distribution of $\varepsilon^{(i)}_{\text{UKF}} - \varepsilon^{(i)}_{\text{SE}}$ in Figure 7.2. The third interquartile corresponding to $Q_W = 1.1$ is excluded because the plot could not be scaled appropriately. For the position, the value is $77m$. For the velocity, 67m/s, and for the turn rate, 7.8 degrees/s.

Choice of basis functions made minimal difference in this experiment. We reran the experiment using $N = 8$ Haar wavelet functions instead of sinusoidal basis functions. Results for the most nonlinear setting $Q_W = 1.1$ are shown in Table 7.3. Filtering errors for both sets of basis functions were close to one another. This is because the Gaussian approximation and tuning parameters of the unscented transform

Q_W	.1	.3	.5
RMSE UKF (divs)	49.9 m (1)	49.7 m (3)	55.0 m (12)
RMSE SE-UKF (divs)	49.9 m (1)	49.8 m (4)	56.4 m (7)

Q_W	.7	.9	1.1
RMSE UKF (divs)	66.9 m (28)	92.2 m (75)	136.7 m (107)
RMSE SE-UKF (divs)	63.4 m (17)	71.5 m (20)	83.5 m (50)

Table 7.1: Mean position errors and divergences for 1000 runs of the filter. Larger values of Q_W result in more erratic trajectories. The filter was deemed to have diverged if the position error was greater than 1km, or if the filter failed due to the appearance of a non-positive definite covariance matrix. Divergent runs were not included in the average. The number of divergences is reported in parentheses

Q_W	.1	.3	.5
Runtime UKF	9s	26s	43s
Runtime SE-UKF	26s	29s	27s

Q_W	.7	.9	1.1
Runtime UKF	61s	80s	96s
Runtime SE-UKF	26s	27s	26s

Table 7.2: Runtimes for the filtering algorithms under varying degrees of nonlinearity. The principal determinant of runtime for the standard filter is the number of steps in the discretisation. We find this must increase as nonlinearity increases in order to keep the algorithm stable. The SE-UPF uses an out-of-the-box adaptive runge-kutta solver which appears to be stable under all settings.

have a larger effect on the filter than specifics of the series expansion approximation.

Surprisingly, we found that choosing the symmetric square root of P_t (that is, the matrix that satisfies $\mathbf{S}^2 = \mathbf{P}_t$, implemented in MATLAB as sqrtm()) instead of the Cholesky decomposition improved the accuracy of our algorithm considerably (though this choice did not improve performance of the standard UKF). The choice of matrix square root is known to affect fourth-order and higher terms in the Taylor expansion

Basis	Pos. Error	Vel. Error	Turn Error
Sine	53.4 m	20.8 m/s	0.300 deg/s
Haar	53.6 m	20.9 m/s	0.301 deg/s

Table 7.3: Error induced by using a Haar wavelet basis versus error from a sinusoidal basis. Median error from 1000 runs of the filter. We used the most highly nonlinear setting, $Q_W = 1.1$.

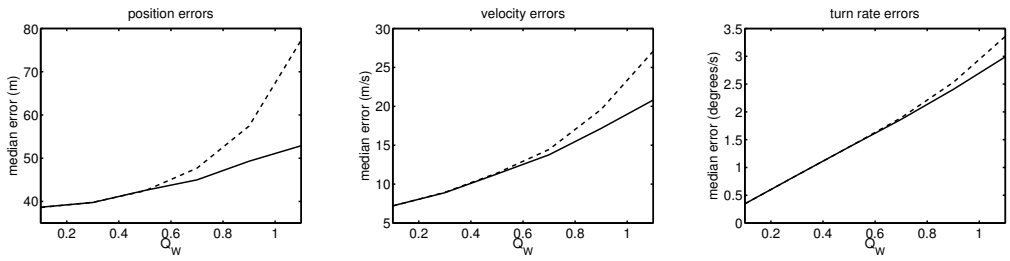


Figure 7.1: The x-axis shows the diffusion coefficient Q_W of the Brownian motion driving $\mathbf{X}_7(t)$. We use this as a measure of the nonlinearity of the system. For a range of values of Q_W, we simulated 1000 trajectories of the signal, observed with noise. We plot median values of the error for the unscented Kalman filter (dotted line) and series expansion filter (solid line).

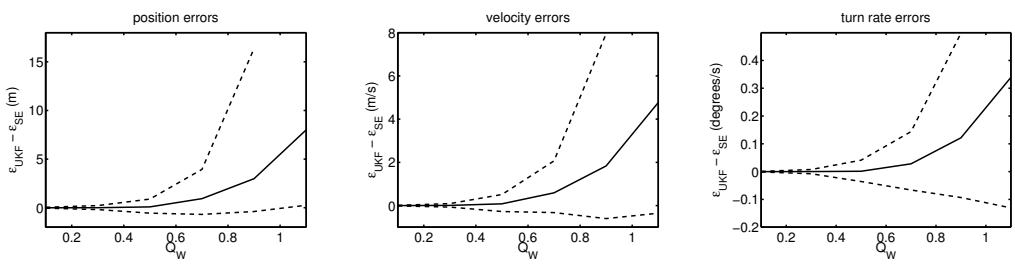


Figure 7.2: The x-axis shows the diffusion coefficient Q_W of the Brownian motion driving $\mathbf{X}_7(t)$. We use this as a measure of the nonlinearity of the system. For a range of values of Q_W, we simulated 1000 trajectories of the signal, observed with noise. We plot median values of the difference in error between the unscented Kalman filter and series expansion Kalman filter (solid line), together with the first and third quartiles (dashed lines). Errors were computed seperately for position, velocity and turn rate of the aircraft. The last point in the upper range is omitted because its inclusion would skew the scaling in the image. Values for these points can be found in Section 7.2.1.

of the transition function $\mathbf{f}$ [31]. This is in agreement with our intuition: the transition function in the UKF is locally linear, and hence can be approximated with a low-order Taylor series. On the other hand, the series expansion filter uses a more nonlinear transition function and one must consider higher order terms.

7.2.2 Series expansion step size

In the prediction step of the standard unscented filter, one discretises the process $\mathbf{X}$, and iteratively applies the unscented transform at each timestep. The aim is to estimate the mean and covariance of $\mathbf{X}_t$ at some time t, given an appropriate initial condition. Repeated applications of the unscented transform at each timestep induce error in this estimate. We will refer to error of this nature as 'projection error'.

On the other hand, the error in the SE-UKF comes from the error induced by the series expansion approximation, coupled with the error induced by a single application of the unscented transform. Error also accrues from numerical solution of the ODE, but in our experiment, this is negligible compared to other sources of error. Empirically, we observe that the accuracy of the series expansion approach improves with the number N of basis functions that we use, and deteriorates with the time T between observations. We will refer to error induced by the series expansion as 'approximation error'.

In one sense, these two approaches represent two extremes of a more general framework. For example, we might use the series expansion approximation to estimate the mean and variance of $\mathbf{X}_{T/2}$. We could then form a Gaussian approximation of its distribution, and use this as the initial condition (starting at time $T/2$) for a second application of the series expansion trick to estimate the mean and covariance of $\mathbf{X}_T$. In effect, we reduce the approximation error at the cost of increasing the projection error.

In order to investigate the effect of trading approximation error for projection error, we ran the filtering experiment of Section 7.2.1 using the most nonlinear setting, $Q_W = 1.1$. Recall that the time interval between observations was $T = 8$ seconds. We divided this interval into K subintervals of length T/K. At the end of each subinterval, we re-initialised the series expansion approximation, using as initial condition the mean and variance computed at the previous sub-interval.

Table 7.4 shows that one can reduce the error slightly by repeatedly employing the series expansion approximation over a shorter timescale, thus trading approximation error for projection error. As the number K of projections becomes large, the error grows to match that of the standard UKF.

K	Pos. Error	Vel. Error	Turn Error
1	54.0 m	21.1 m/s	0.293 deg/s
2	51.6 m	20.2 m/s	0.291 deg/s
4	51.6 m	20.6 m/s	0.290 deg/s
8	55.5 m	21.6 m/s	0.294 deg/s
16	61.7 m	22.8 m/s	0.304 deg/s
32	69.3 m	24.2 m/s	0.322 deg/s

Table 7.4: The effect of trading approximation error for projection error. Median errors over 1000 runs of the filter. Rows are indexed by number K of projections per observation. We used the most challenging parametrisation $Q_W = 1.1$ to generate the data. Observe that results for $K = 32$ correspond closely to the errors for the standard UKF in Figure 7.1.

7.3 Discussion and conclusions

In this chapter, we have presented a Gaussian filter based on the series expansion approximation. The novel contributions of this paper focus on improving the predictive distribution, so it is straightforward to construct a smoother using similar methods: for example one can use the unscented smoother [89] or Gaussian smoother [101] directly.

Two questions follow naturally from this work. Firstly, how does one choose parameters for the unscented transform in a sensible way? Secondly, what basis functions should one use in the series expansion? In most cases the optimal solution for either question is likely to be very difficult to compute.

All filters based around the unscented transform must somehow deal with the first issue. Various heuristics can be found in the literature on how one might choose the tuning parameters: see, for example [102], [103]. In some cases, a poor choice of tuning parameters can cause the covariance matrix in the prediction step to fail to be positive definite. This causes the filter to diverge.

When using a common set of tuning parameters ($\alpha = 1, \kappa = 3 - n, \beta = 2$, where n is the dimensionality of the system [104]), we found the matrix degeneracy problem to occur in both the series expansion filter (about 1% of runs) and the standard unscented filter (about 10% of runs). This is a known issue when using these settings in a high-dimensional context [104]. We found that increasing κ slightly to $\kappa = 5 - n$ in the series expansion filter removed the divergence issue without affecting performance. On the

other hand, the cubature Kalman filter settings ($\alpha = 1, \kappa = 0, \beta = 0$) performed poorly for the series expansion filter. This is because the higher dimensionality of the system causes the sigma points to be spread far out from the mode (that is, $\lambda = \alpha^2(n+\kappa) - n$ is large).

We also compared our algorithm to the third-order Gauss-Hermite Kalman filter (GHKF). This algorithm also exhibited numerical instability, with the predicted covariance matrix often failing to be positive definite. When we discarded test runs on which the GHKF diverged, we found that our algorithm performed comparably to the GHKF. This is despite the fact that the cost of the GHKF scales exponentially with dimension. In the present setting, the GHKF used $3^7 = 2187$ sigma points, and required several days of computation time to perform a comparison for a single value of Q_W.

We now address the issue of the choice of orthonormal basis. We performed the same filtering experiments using a sinusoidal basis, and a basis of Haar wavelets. Results were similar in both cases. Our explanation for this is that we already induce significant error by assuming the filtering distribution is Gaussian. This error is significantly larger than the error induced by the series expansion approximation, so the latter error is difficult to detect.

As we showed in Chapter 5.1, it is possible to set the series expansion approximation error to 0 in certain linear inference problems. One plausible heuristic for choosing a set of basis functions in the nonlinear setting is to construct a linear approximation to the problem. One then computes the optimal basis functions for the linearized dynamics as in Chapter 5.1. However, in our numerical tests we found that these basis functions were prone to numerical instability, and furthermore they do not come with a guarantee of uniform convergence. We note that this may be a useful strategy in filtering problems that are 'almost' linear.

Chapter 8

The series expansion unscented Particle filter

As we saw in Chapter 3, the *particle filter* is a promising candidate for solving low- and medium-dimensional nonlinear filtering problems. In this chapter, we discuss a novel way of applying the series expansion approximation to construct better importance distributions for use in a particle filter.

Recall that the bootstrap filter uses the prior dynamics of the signal to propagate a collection of particles that approximate the filtering distribution forward in time. One can often hope to improve upon the bootstrap filter by using better proposal distributions. However, when the underlying signal is a diffusion process, it is often impossible to compute the density $p(\mathbf{X}_{t_k}|\mathbf{X}_{t_{k-1}})$. For this reason, the importance distributions that one can employ are rather restricted, since in most cases one must rely on some sort of cancellation from the importance distribution.

In this chapter, we use the series expansion approximation to interpret the diffusion process at time t_k as the image of a Gaussian distribution under a certain nonlinear transform f. That is, $\mathbf{X}_{t_k} \approx f(\mathbf{X}_{t_{k-1}}, \mathbf{Z})$. We use the unscented transform to construct an importance variate $\mathbf{V}$ such that $f(\mathbf{X}_{t_{k-1}}, \mathbf{V})$ sidesteps the particle degeneracy issue while still producing tractable importance weights.

A number of other methods have been applied to the problem. Fearnhead et al. [44] [45] use a form of rejection sampling to construct an unbiased estimator of the importance weights. This obviates the need for some term in the importance distribution to cancel with $p(\mathbf{X}_{t_k}|\mathbf{X}_{t_{k-1}})$, allowing the authors to employ more general importance distributions. However, the algorithm is only applicable to certain classes of diffusion process, and it only performs well when the variance of the estimator is sufficiently

low.

Rimmer et al. [83] discuss the use of a particle filter for maximum likelihood estimation of some parameter governing the dynamics of the diffusion process. Use of the unscented transform within a discrete approximation to the diffusion is discussed. Our use of the unscented transform is different, and considerably cheaper in toerms of computational cost.

Murray and Storkey [90] use a high-order numerical integration scheme to aproximate both filtering and smoothing densities by means of a forward-backward recursion. Practical applications of the particle filter to fMRI data are discussed in [6].

Other approaches typically use the Girsanov theorem to construct importance processes that have a tractable density with respect to the law of the signal $\mathbf{X}$. Maroulas and Stinis [105] use 'drift homotopy' (a method related to simulated annealing) to find 'good' importance processes. Särkkä and Sottinen [89] suggest linearising the SDE to construct an importance process with the appropriate mean and covariance.

The rest of the chapter is structured as follows. In Section 8.1 we show how to combine importance sampling with the series expansion approximation. Section 8.2 provides some numerical experiments, and we discuss our findings and some possibilities for future work in Section 8.3.

8.1 The series expansion unscented particle filter

In Chapter 5.1, we saw that the value of $\mathbf{X}_{t_k}$ can be approximated as the image of a collection of i.i.d. Gaussian random variables (along with a possibly random initial condition) under a nonlinear transform. That is,

$$\hat{\mathbf{X}}_t = \mathbf{f}(t, \mathbf{X}_0, \mathbf{Z}_1, \ldots, \mathbf{Z}_N). \tag{8.1}$$

For the purposes of describing the SE-UPF, we assume the initial condition is fixed, though this can easily be generalised.

One is often interested in computing expectations of the form

$$\mathbb{E}[g(\mathbf{X}_{t_k})] \approx \mathbb{E}[g(\hat{\mathbf{X}}_{t_k})] = \mathbb{E}\left[g\left(f(T, \mathbf{X}_{t_{k-1}}, \mathbf{Z}_{1:N})\right)\right], \tag{8.2}$$

where f is the solution of the ODE (5.10) described in Section 5.1.1.

In cases where the expectation cannot be computed exactly, one natural way to proceed is to draw a number of i.i.d samples $\{\hat{\mathbf{X}}^i_{t_k}\}_{i \leq n}$ from $\hat{\mathbf{X}}_{t_k}$ and approximate the

expectation via the Monte-Carlo estimate

$$\begin{aligned}\mathbb{E}[g(\hat{\mathbf{X}}_{t_k})] &\approx \frac{1}{n}\sum_{i=1}^{n} g(\hat{\mathbf{X}}^i_{t_k}) \\ &= \frac{1}{n}\sum_{i=1}^{n} g\left(f(T, \mathbf{X}_{t_{k-1}}, \mathbf{Z}^i_{1:N})\right) \end{aligned} \tag{8.3}$$

The right-hand side of (8.3) is an unbiased estimator of $\mathbb{E}[g(\hat{\mathbf{X}}_{t_k})]$. However, its variance may be unacceptably high. Consider, for example, a function g that takes a large value somewhere in the tails of the distribution of $\hat{\mathbf{X}}_{t_k}$.

To work around this issue, one can attempt to construct a more suitable collection of 'importance' random variables $\mathbf{Z}_{1:N} \sim q(\cdot)$. We can perform importance sampling by drawing i.i.d samples $\{\mathbf{Z}^i_{1:N}\}$ from $q(\cdot)$ and weighting the Monte-Carlo estimate appropriately:

$$\mathbb{E}[g(\hat{\mathbf{X}}_{t_k})] \approx \frac{1}{n}\sum_{i=1}^{n} g\left(f(t, \mathbf{X}_{t_{k-1}}, \mathbf{Z}^i_{1:N})\right) \frac{\mathcal{N}(\mathbf{Z}^i_{1:N}|0, \mathbf{I}_{Nd\times Nd})}{q(\mathbf{Z}^i_{1:N})}. \tag{8.4}$$

If the distribution q is chosen appropriately, one can achieve a reduction in the variance of the estimator. We will use this methodology to 'guide' particles in the SE-UPF toward regions of high likelihood.

We will now describe one way of applying these ideas in the context of particle filtering. For a general nonlinear diffusion, it is not feasible to compute the transition density $p(\mathbf{X}_{t_k}|\mathbf{X}_{t_{k-1}})$. Instead of working directly with the density of $\mathbf{X}_{t_k}$, we opt instead to use importance sampling to modify the driving noise. In terms of the SDE approximation derived in Section 5.1.1, the expression for the unnormalised weight update (3.55) is

$$\tilde{w}^i_{t_k} = w^i_{t_{k-1}} \frac{p(\mathbf{Y}_{t_k}|f(t, \mathbf{X}^i_{t_{k-1}}, \mathbf{Z}^i_{1:N}))\mathcal{N}(\mathbf{Z}^i_{1:N}|0, \mathbf{I}_{Nd\times Nd})}{q(\mathbf{Z}^i_{1:N}|\mathbf{X}^i_{t_{k-1}}, \mathbf{Y}_{t_k})}, \tag{8.5}$$

Here, q is playing the role of the importance distribution in (3.55), though we have modfied the driving noise of $\mathbf{X}$ instead of modifying the distribution of $\mathbf{X}_{t_k}$ directly. As we will see in the next section, the unscented transform provides a computationally inexpensive means of constructing useful importance distributions for the driving noise.

8.1.1 Choice of importance distribution

There are several ways in which one could form an appropriate importance distribution. One possible way to proceed is to construct a Gaussian approximation of the joint

distribution $p(\mathbf{Y}_{t_k}, \mathbf{Z}_{1:N} | \mathbf{X}_{t_{k-1}})$. Given such an approximation, standard results about the multivariate normal distribution allow us to compute a Gaussian approximation to $p(\mathbf{Z}_{1:N} | \mathbf{Y}_{t_k}, \mathbf{X}_{t_{k-1}})$, which we will use as our importance distribution $q(\cdot)$.

In what follows, we will use the unscented transform to construct a Gaussian approximation of the joint distribution of $\mathbf{Y}_{t_k}$ and $\mathbf{Z}_{1:N}$. The random variable $\mathbf{Z}_{1:N}$ is $N \times d$-dimensional standard normal. We select $2Nd+1$ sigma points $\{\sigma^i\}$ to capture the mean and covariance of $\mathbf{Z}_{1:N}$.

For a suitable initial condition $\mathbf{X}^{\text{ic}}$ (e.g. the ensemble mean, or the location of a given particle $\mathbf{X}^i_{t_{k-1}}$), we set

$$\mathcal{Y}^j = h(f(\mathbf{X}^{\text{ic}}, \sigma^j)), \qquad 1 \leq j \leq 2Nd+1. \tag{8.6}$$

We can apply equations (3.38), (3.39) and (3.40) with the appropriate weighting to find μ, $\mathbf{S}$ and $\mathbf{C}$ (recall that these quantities approximate the mean of $\mathbf{Y}$, the covariance of $\mathbf{Y}$ neglecting observation noise, and the cross-covariance of $\mathbf{Z}_{1:N}$ and $\mathbf{Y}$ respectively). When we account for the additional variance $\mathbf{R}$ added by the observation noise, we obtain

$$\text{Cov}[(\mathbf{Y}_{t_k}, \mathbf{Z}_{1:n}) \mid \mathbf{X}^{\text{ic}}] \approx \begin{pmatrix} \mathbf{S}+\mathbf{R} & \mathbf{C}^\top \\ \mathbf{C} & \mathbf{I}_N \end{pmatrix}. \tag{8.7}$$

The approximate mean and variance of $\mathbf{Z}_{1:N}$ conditional on $\mathbf{Y}_{t_k}$ is then

$$\mathbb{E}[\mathbf{Z}_{1:N} | \mathbf{Y}_{t_k}, \mathbf{X}^{\text{ic}}] \simeq \mu_{Z|Y} = \mathbf{C}(\mathbf{S}+\mathbf{R})^{-1}(\mathbf{Y}_{t_k} - \mu), \tag{8.8}$$

$$\text{Cov}[\mathbf{Z}_{1:N} | \mathbf{Y}_{t_k}, \mathbf{X}^{\text{ic}}] \simeq \Sigma_{Z|Y} = \mathbf{I}_N - \mathbf{C}(\mathbf{S}+\mathbf{R})^{-1}\mathbf{C}^\top. \tag{8.9}$$

We can use these quantities as a basis for constructing an importance distribution q. The simplest method is simply to set $q = \mathcal{N}(\mu_{Z|Y}, \Sigma_{Z|Y})$. However, theoretical analysis of importance sampling shows that one should employ a distribution with heavier tails than the target distribution. One possibility is to set the diagonal entries of $\Sigma_{Z|Y}$ to 1.

In order to compute $\mu_{Z|Y}$ and $\Sigma_{Z|Y}$, we must solve $2Nd+1$ differential equations: one for each sigma point. This can be done for the ensemble of particles as a whole, or for each unique particle individually. That is, $\mu_{Z|Y}$ and $\Sigma_{Z|Y}$ may be computed 'globally' using some summary statistic of the ensemble, or 'locally', depending on each individual particle. For the 'global' approach, we use the ensemble mean as an initial condition in (8.6). This represents an increase in cost over the bootstrap filter

that is independent of the number of particles. For the local approach, we use the location of each of the n particles as the initial condition in (8.6).

The 'local' method requires that (8.6) be solved for each individual particle. Naively, one might think that this entails solving the ODE with n distinct initial conditions. However, after resampling, many of the particles share the same location. In practice, we need to solve significantly fewer than n systems of ODEs. The increase in performance cost of the 'local' method is smallest in situations that standard particle filters struggle to cope with due to low effective sample size.

In practice, we find that the posterior mean and covariance tend only to change for the first few coefficients $\mathbf{Z}_{1:M}$, with the distribution of the remainder being similar to the prior (i.e. standard normal). Intuitively speaking, this is sensible. The 'high frequency' basis functions control the behaviour of the SDE at small length-scales. In the typical case, one should not expect to be able to deduce small-scale behaviour from observations spaced far apart in time.

Since observations tend not to be informative about higher-frequency components, we can perform importance sampling guided by the unscented transform on a small number of coefficients $\mathbf{Z}_{1:M}$, and draw the remaining coefficients $\mathbf{Z}_{M+1:N}$ from the prior. The value of M depends on the nature of the model, but in the high-dimensional experiment below, we saw good results with $M = 2$, $N = 4$.

8.2 Numerical experiments

8.2.1 Sinusoidal diffusion

For our first test, we reproduced the setup from one of the experiments in [44]. A sample path was drawn from the one-dimensional diffusion

$$d\mathbf{X}_t = \sin(\mathbf{X}_t)dt + d\mathbf{W}_t, \mathbf{X}_0 = 0. \tag{8.10}$$

We made 50 noisy observations of the process, spaced $T = 1$ units of time apart. The variance of the observation noise was set to $\mathbf{R} = .5$. The performance of the SE-UPF was compared against that of the bootstrap filter (using an Euler-Maruyama discretisation) and the random-weight particle filter from [44]. The importance distribution for the RWPF used the ensemble mean (i.e. the same distribution was used for each particle). We assess the performance of the filters by measuring the variance their estimate of the filter mean over several independent runs, having fixed a sample path and a set of observations. We assume that filters with a lower variance are performing better.

It is not straightforward to devise a simple and fair way of comparing these filters. The random weight filter makes an unbiased estimate of the importance sampling weights, whereas both the SE-UPF and Euler-Maruyama bootstrap filter are subject to approximation bias. Computational cost per particle is not necessarily an ideal metric. One can choose a large timestep in an Euler scheme or when solving (5.10) to propagate particles forward in time cheaply. Provided that one is prepared to tolerate this bias, these methods will dominate the RWPF. In addition, one or other algorithm could be heavily optimised to boost its performance.

Another possibility is to use a set number of particles and compare the performance across algorithms without regard to computational cost. Again, this is not ideal since any useful improvement to a particle filter should be expected to outperform the baseline method of simply adding more particles to the bootstrap filter up to a computational cost equivalent to that of the improvement.

For sufficiently low observation noise (roughly $\mathbf{R} < .3$), the RWPF outperformed the SE-UPF and Euler-Maruyama bootstrap filters for a fixed number of particles ($n = 500$). However, the RWPF was more computationally expensive. For $\mathbf{R} > .3$, the SE-UPF outperformed the RWPF and bootstrap filters on a per-particle basis. It was cheaper than the RWPF, and the bootstrap filter. It is surprising that the SE-UPF would be cheaper than the bootstrap filter. This is because the standard Euler-Maruyama scheme needs to generate draws from a standard normal distribution for each timestep, whereas the SE-UPF needs fewer such draws. The relative variance of the estimates of the filter mean are shown in Figure 8.1.

RMSE	$\mathbf{R} = .5$	$\mathbf{R} = .25$
Series expansion	.387	.28
Bootstrap	.387	.278
Exact	.3869	.274

Table 8.1: Root mean squared errors for the filters. The figure for the exact filter is lowest: this is not surprising since the sampling mechanism is unbiased.

8.2.2 Coordinated turn model

For our next experiment, we simulated a number of sample paths from a model of an aircraft performing a 'coordinated turn' as in Chapter 5.1 and Chapter 7.

Time average of variance	$\mathbf{R} = .5$	$\mathbf{R} = .25$
Series expansion	.00039	.0002
Bootstrap	.0006	.0033
Exact	.00047	.00016

Table 8.2: Summary of Figure 8.1. Average over the variance of the filter estimates at each point in time. In the high-noise regime, the series expansion filter outperforms the others, while in the low-noise regime the exact sampling algorithm outperforms the others.

Approximate runtime (seconds)	
Series expansion	4
Bootstrap	7
Exact	12

Table 8.3: Representative runtimes for each algorithm for 50 observations over one iteration of the filter.

The settings used in the SE-UKF algorithm of Chapter 7 assumed that the observations of the azimuthal and attitudinal angle were very precise. Particle filters cannot usually cope with high process noise and low observation noise simultaneously. In such a case, all particles but one are typically assigned very low weights, with the result that the filtering distribution is approximated by a delta function.

Both the series expansion filter and the bootstrap filter diverged almost immediately under the settings of Chapter 7. For this reason, we use the alternative settings $X_{t_0} = (1000, 100, 1000, 100, 1000, 100, 600)$, $\mathbf{Q} = \mathrm{diag}(100, 100, 100, 1000)$. The inter-observation time was set to $T = 1$. As before, we modelled observations as arriving through a ground-based radar dish that can determine range, azimuthal angle, and attitudinal angle. The observation error was assumed to have covariance $\mathbf{R} = \mathrm{diag}([1, .005, .005])$. In this case, the observation noise is considerably larger than in Chapter 7. Due to the reduced inter-observation time, the signal dynamics are less variable. This new system is much more amenable to particle filtering than that the parameterisations discussed in Chapter 7.

We tested the SE-UPF against the bootstrap filter. The RWPF is not applicable in this situation since the diffusion matrix is not of full rank. To be as fair as possible, we

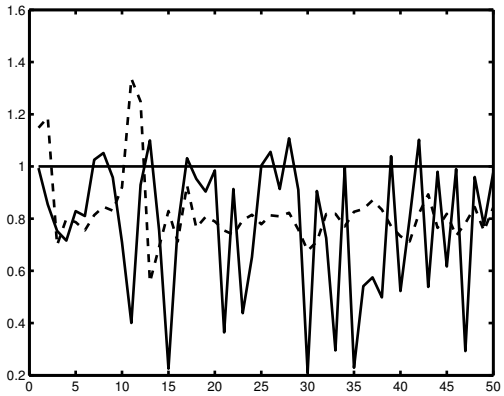


Figure 8.1: Variance of the SE-UPF estimate of the filter mean divided by the variance of the RWPF estimate (broken line) and the variance of the bootstrap filter estimate (solid line).

used the same code for the bootstrap filter as for the SE-UPF. The bootstrap filter code was modified to draw $\mathbf{Z}_{1:N}$ from the prior distribution instead of using an importance distribution.

Both tests used approximately the same amount of processor time: for the standard particle filter, we used $n = 450$ particles. For the series expansion filter, we used $n = 200$ particles. We drew a sample path from $\mathbf{X}$ and ran both filters 500 times. for each filter, we computed $\mathrm{Var}(\hat{m}_t)$, the variance in the estimate of the filtering mean at time t. A large variance between runs of a filter indicates that it is a poor approximation of the optimal filter.

We computed $\mathrm{Var}(\hat{m}_t^{\mathrm{SE}})$ and $\mathrm{Var}(\hat{m}_t^{\mathrm{B}})$, respectively the sample variance in the series expansion filter and bootstrap filter estimate of $\mathbb{E}[\mathbf{X}_t|\mathbf{Y}_{1:t}]$. In Figure 8.3, we plot $\mathrm{Var}(\hat{m}_t^{\mathrm{SE}})/\mathrm{Var}(\hat{m}_t^{\mathrm{B}})$ as a function of time. We found that the variances were comparable initially, but as time progressed the variance of the bootstrap filter estimate grew markedly in comparison to the variance of the series expansion filter.

In Figure 8.4, we show normalised histograms that display the root mean squared error from each run of both filters. The error in the position and velocity of the SE-UPF estimate is noticeably lower than that of the bootstrap filter.

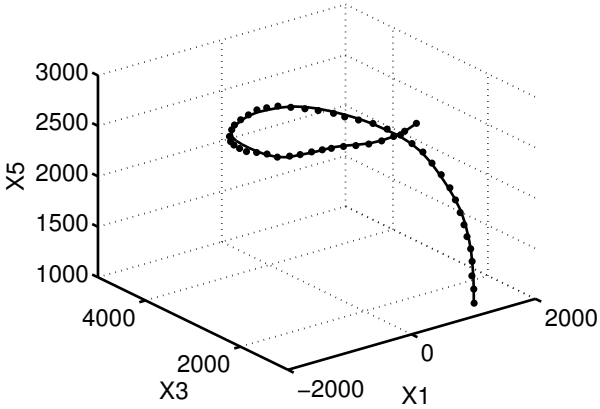
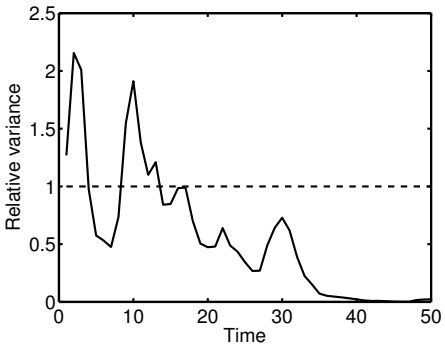
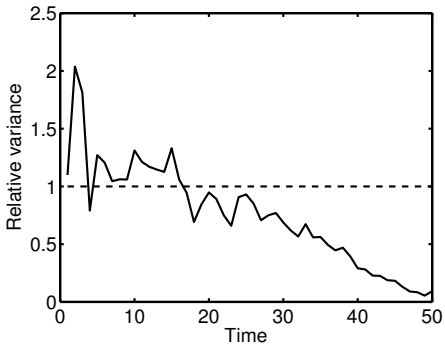


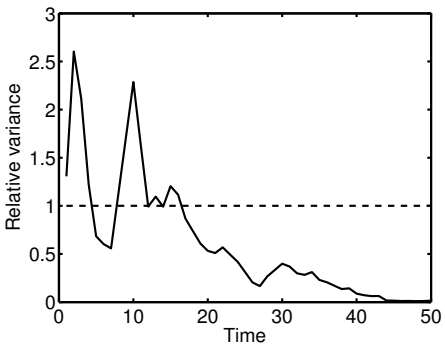
Figure 8.2: Plot of the position components from a sample path of the coordinated turn model, with the mean position as estimated by the SE-UPF (black dots).



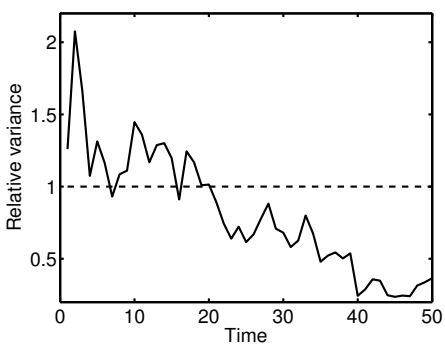
(a) Relative variance of the filter mean for X_1



(b) Relative variance of the filter mean for X_2

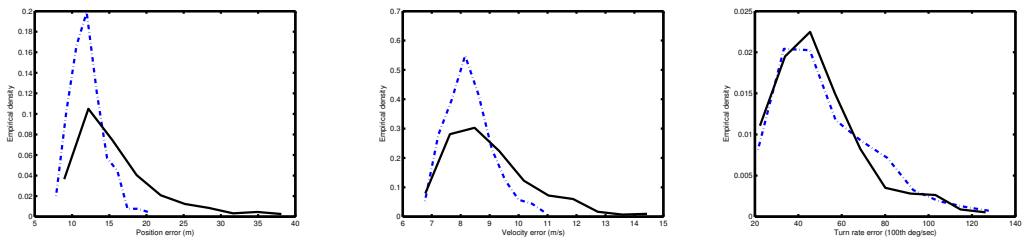


(c) Relative variance of the filter mean for X_3



(d) Relative variance of the filter mean for X_4

Figure 8.3: Relative performance of bootstrap and series expansion filters for the coordinated turn model experiment



(a) Position (b) Velocity (c) Turn rate

Figure 8.4: RMSE for position, velocity, and turn rate estimates from both filters. The SE-UPF error (broken line) tends to be smaller than the bootstrap error (solid line).

8.3 Discussion

Our experiments suggest that the SE-UPF is more robust than the bootstrap filter. In some scenarios it also outperforms the random weight particle filter, while being more generally applicable. The methodology of section 5.1 suggests some interesting possibilities. It may be possible to use heavy-tailed importance distributions in order to provide theoretical guarantees on the variance of the importance weights.

Given the improvement of the SE-UPF over the bootstrap filter, it would be interesting to investigate the application of series expansion importance sampling to particle Markov chain Monte-Carlo (PMCMC) methods. The estimation of parameters of a diffusion process is a difficult task in general, but PMCMC methods are a a promising tool for this problem. However, the bootstrap filter can struggle to cope within PMCMC. The filter will be run repeatedly, and the particles will often use dynamics that are very different from the dynamics that generated the observations. As a result, most observations will behave like outliers, and the effective sample size of the filter will be lower.

The SE-UPF can direct particles to ‘interesting’ areas, and has been shown to have a lower variance than the bootstrap filter. For this reason, we expect that it will improve the performance of a PMCMC algorithm.

Chapter 9

Conclusion

In this work, we have investigated some of the possibilities offered by incorporating a 'Fourier series perspective' on Brownian motion into the theory of stochastic differential equations. The idea applying Fourier analysis to the theory of SDEs via a decomposition of Brownian motion is not new. Series expansion approximations have previously been used to approximate solutions of stochastic partial differential equations (see [73], [106]).

However, the Fourier series perspective is not commonplace in the physics or stochastic analysis literature. Most standard textbooks on the theory of diffusion processes (e.g. [21], [9], [10]) do not mention the Fourier series perspective of Brownian motion (though the Karhunen-Lóeve expansion is described in [22] without reference to its application in the approximation of SDEs).

The contribution of this thesis was to explore some novel applications of the Fourier series perspective of stochastic differential equations. In Chapter 6 we showed that the Fourier coefficients of a white-noise expansion can be exploited to gain fine -grained control over the behaviour of a Brownian sample path. This increased level of control was used to good effect in a Metropolis-hastings algorithm, where we were able to generate proposals that behaved like our sample data.

In Chapter 7 we described a novel interpretation of the time-t value of a SDE as the 'image' of an N-dimensional gaussian under a nonlinear transform. The components of the Gaussian distribution were given by the Fourier coefficients of the driving Brownian motion. We found that one needs a relatively small number of components in order to get a good approximation to the true distribution of the SDE at time t - in effect, providing a dimensionality reduction. The reduction in the number of covariates allowed us to apply the unscented transform to construct a Gaussian approximation of

the target distribution in an efficient manner.

In Chapter 8, we showed that one can conduct importance sampling on the Fourier coefficients of a Brownian motion. We used this technique to 'guide' a set of particles in a particle filter towards a region of high likelihood, as determined by a Gaussian approximation. We showed that the resultant filter can outpeform existing filters.

A number of technical questions on the subject remain open. For example, it would be useful to obtain quantitative bounds on the error induced by the series expansion approximation under weak assumptions on the basis functions $\{\phi_i\}$. This would be useful for practical applications. However, it would also preclude the possibility of encountering numerical issues that are sometimes encountered in Fourier analysis. For example, the *Gibbs phenomenon* occurs when one attempts to decompose a square wave as a series of sine waves. There are certain points on the sinusoidal approximation that do not converge to points on the square wave, *no matter how many terms are used*. The point here is that L^2 convergence does not necessarily imply pointwise convergence. In this work, we have concentrated on the value of a diffusion process $\mathbf{X}$ *at a specific time* t. In other words, we rely on pointwise convergence.

We expect that rigorous analysis of the series expansion approximation will be difficult. The issue of convergence was studied in some depth and generality by Wong and Zakai [74], McShane [75] and others with no decisive resolution: sufficient conditions can be found in [107] though these are restrictive.

We do not claim that our methods are the last word on the subject, or that we have exhausted all possibilities for exploiting the series expansion approximation. Indeed, we argue that the most important contribution of this thesis is not the specific details of the methods we have described. Rather, it is that there is *something here that may have been overlooked*, and that it may be important.

‘Take it from me, there’s nothing like a job well done.
Except the quiet enveloping darkness at the bottom of a bottle of Jim Beam after a job done any way at all.’

– *Hunter S. Thompson*

Bibliography

[1] E.P. Odum. *Fundamentals of ecology*. Thomson-Brooks/Cole, 1953.

[2] R.C. Merton. Theory of rational option pricing. *The Bell Journal of Economics and Management Science*, 4:141–183, 1973.

[3] D.T. Gillespie. The chemical Langevin equation. *Journal of Chemical Physics*, 113,1:297–306, 2000.

[4] G. Kallianpur. Weak convergence of stochastic neuronal models. *Stochastic Methods in Biology*, 70:116–145, 1987.

[5] H.A. Dijkstra, L.M. Frankcombe, and A.S. von der Heydt. A stochastic dynamical systems view of the Atlantic Multidecadal Oscillation. *Philosophical Transactions of the Royal Society A*, 366:2543–2558, 2008.

[6] L. Murray and A. Storkey. Continuous time particle filtering for fMRI. *Advances in Neural Information Processing Systems*, 20:1049–1056, 2008.

[7] J. Daunizeau, K.J. Friston, and S.J. Kiebel. Variational Bayesian identification and prediction of stochastic nonlinear dynamic causal models. *Physica D*, pages 2089–2118, 2009.

[8] W. Feller. *An Introduction to Probability Theory and its Applications, Volume II*. Wiley, 1971.

[9] I. Karatzas and S.E. Shreve. *Brownian Motion and Stochastic Calculus*. Springer, 1991.

[10] B. Öksendal. *Stochastic Differential Equations*. Springer, 2007.

[11] Robert Brown. Xxvii. a brief account of microscopical observations made in the months of june, july and august 1827, on the particles contained in the pollen of

plants; and on the general existence of active molecules in organic and inorganic bodies. *The Philosophical Magazine, or Annals of Chemistry, Mathematics, Astronomy, Natural History and General Science*, 4(21):161–173, 1828.

[12] E. Nelson. *Dynamical theories of Brownian motion*, volume 17. Princeton university press Princeton, 1967.

[13] Nicolas E. Humphries, Nuno Queiroz, Jennifer R. M. Dyer, Nicolas G. Pade, Michael K. Musyl, Kurt M. Schaefer, Daniel W. Fuller, Juerg M. Brunnschweiler, Thomas K. Doyle, Jonathan D. R. Houghton, Graeme C. Hays, Catherine S. Jones, Leslie R. Noble, Victoria J. Wearmouth, Emily J. Southall, and David W. Sims. Environmental context explains Lévy and Brownian movement patterns of marine predators. *Nature*, 465(7301):1066–1069, June 2010.

[14] M. J. Reid and A. Brunthaler. The proper motion of sagittarius a*. ii. the mass of sagittarius a*. *The Astrophysical Journal*, 616(2):872, 2004.

[15] L.C.G. Rogers and D. Williams. *Diffusions, Markov processes, and martingales*, volume 2. Cambridge Univ Pr, 2000.

[16] Erwin Kreyszig. *Introductory functional analysis with applications*, volume 81. Wiley New York, 1989.

[17] Nelson Dunford and Jacob T Schwartz. Linear operators, vol. i. *Interscience, New York*, 1963, 1958.

[18] Simon Lyons, Amos Storkey, and Simo Sarkka. The coloured noise expansion and parameter estimation of diffusion processes. In P. Bartlett, F.C.N. Pereira, C.J.C. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1961–1969. 2012.

[19] S.M.J Lyons, S. Särkkä, and A.J. Storkey. Series expansion approximations of Brownian motion for non-linear Kalman filtering of diffusion processes. *IEEE Transactions on Signal Processing*, 62:1514–1524, 2013.

[20] D. Nualart. *The Malliavin calculus and related topics*. Springer, 1995.

[21] C.W. Gardiner. *Handbook of Stochastic Methods for Physics, Chemistry and the Natural Sciences*. Springer-Verlag, 1983.

[22] P.E. Kloeden and E. Platen. *Numerical Solution of Stochastic Differential Equations*. Springer, 1999.

[23] A. Thiery. Doob h-transforms. `http://linbaba.wordpress.com/2010/06/02/doob-h-transforms/`. Retrieved 18th July 2011.

[24] Cdric Archambeau, Manfred Opper, Yuan Shen, Dan Cornford, and John Shawe-Taylor. Variational inference for diffusion processes. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 17–24. MIT Press, Cambridge, MA, 2008.

[25] Rudolph E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME, Journal of Basic Engineering*, 82:35–45, March 1960.

[26] M.S. Grewal and A.P. Andrews. *Kalman Filtering: Theory and Practice Using MATLAB*. Wiley-IEEE press, 2011.

[27] P. Swerling. First order error propagation in a stagewise differential smoothing procedure for satellite observations. *J. Astronaut. Sci.*, 6:46–52, 1959.

[28] Mohinder S Grewal and Angus P Andrews. Applications of kalman filtering in aerospace 1960 to the present [historical perspectives]. *Control Systems, IEEE*, 30(3):69–78, 2010.

[29] A. Baine and D. Crisan. *Fundamentals of Stochastic Filtering*. Springer, 2009.

[30] Simon J Julier. *Process models for the navigation of high speed land vehicles.* PhD thesis, University of Oxford, 1997.

[31] Simon J. Julier and Jeffrey K. Uhlmann. Unscented filtering and nonlinear estimation. In *Proceedings of the IEEE*, pages 401–422, 2004.

[32] A.H. Jazwinski. *Stochastic Processes and Filtering Theory*, volume 63. Academic Pr, 1970.

[33] S. Särkkä. On unscented Kalman filtering for state estimation of continuous-time nonlinear systems. *IEEE Transactions on Automatic Control*, 52:1631–1641, 2007.

[34] S. Särkkä and J. Sarmavuori. Gaussian filtering and smoothing for continuous-discrete dynamic systems. *Signal Processing*, 93(2):500 – 510, 2013.

[35] Arnaud Doucet and Adam M Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of Nonlinear Filtering*, 12:656–704, 2009.

[36] Nicholas Kantas, Arnaud Doucet, Sumeetpal Sindhu Singh, and Jan Marian Maciejowski. An overview of sequential monte carlo methods for parameter estimation in general state-space models. In *15th IFAC Symposium on System Identification (SYSID), Saint-Malo, France.(invited paper)*, volume 102, page 117, 2009.

[37] O. Cappé, S.J. Godsill, and E. Moulines. An overview of existing methods and recent advances in sequential monte carlo. *Proceedings of the IEEE*, 95(5):899–924, 2007.

[38] Arnaud Doucet, Nando De Freitas, Neil Gordon, et al. *Sequential Monte Carlo methods in practice*, volume 1. Springer New York, 2001.

[39] N.J. Gordon, D.J. Salmond, and A.F.M. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. In *Radar and Signal Processing, IEE Proceedings F*, volume 140, pages 107–113. IET, 1993.

[40] C. Snyder, T. Bengtsson, P. Bickel, and J. Anderson. Obstacles to high-dimensional particle filtering. *Monthly Weather Review*, 136(12):4629–4640, 2008.

[41] Simon J Julier, Jeffrey K Uhlmann, and Hugh F Durrant-Whyte. A new approach for filtering nonlinear systems. In *American Control Conference, 1995. Proceedings of the*, volume 3, pages 1628–1632. IEEE, 1995.

[42] Simon J Julier and Jeffrey K Uhlmann. A new extension of the kalman filter to nonlinear systems. In *Int. symp. aerospace/defense sensing, simul. and controls*, volume 3, pages 3–2. Orlando, FL, 1997.

[43] Rudolph Van Der Merwe, Arnaud Doucet, Nando De Freitas, and Eric Wan. The unscented particle filter. In *NIPS*, pages 584–590, 2000.

[44] P. Fearnhead, O. Papaspiliopoulos, and G.O. Roberts. Particle filters for partially observed diffusions. *Journal of the Royal Statistical Society: Series B*, 70:755–777, 2008.

[45] P. Fearnhead, O. Papaspiliopoulos, G.O. Roberts, and A. Stuart. Random-weight particle filtering of continuous time processes. *Journal of the Royal Statistical Society: Series B*, 72:497–512, 2010.

[46] A Doucet, NJ Gordon, and V Krishnamurthy. Sequential simulation-based estimation of jump markov linear systems. In *Decision and Control, 2000. Proceedings of the 39th IEEE Conference on*, volume 2, pages 1166–1171. IEEE, 2000.

[47] Rudolph van der Merwe, Arnaud Doucet, Nando de Freitas, and Eric Wan. The unscented particle filter. *Advances in Neural Information Processing Systems*, 13, 2000.

[48] H. Sørensen. Parametric inference for diffusion processes observed at discrete points in time: a survey. *International Statistical Review*, 72(3):337–354, 2004.

[49] H. Singer. A survey of estimation methods for stochastic differential equations. In *Proceedings of 6th international conference on social science methodology, Amsterdam*, 2004.

[50] Stefano Maria Iacus. *Simulation and inference for stochastic differential equations: with R examples*. Springer, 2008.

[51] Didier Dacunha-Castelle and Danielle Florens-Zmirou. Estimation of the coefficients of a diffusion from discrete observations. *Stochastics: An International Journal of Probability and Stochastic Processes*, 19(4):263–284, 1986.

[52] Danielle Florens-Zmirou. On estimating the diffusion coefficient from discrete observations. *Journal of applied probability*, pages 790–804, 1993.

[53] Florian Stimberg, Manfred Opper, Guido Sanguinetti, and Andreas Ruttor. Inference in continuous-time change-point models. In *Advances in Neural Information Processing Systems*, pages 2717–2725, 2011.

[54] P. Fearnhead. Computational methods for complex stochastic systems: a review of some alternatives to mcmc. *Statistics and Computing*, 18(2):151–171, 2008.

[55] O. Elerian, S. Chib, and N. Shephard. Likelihood inference for discretely observed nonlinear diffusions. *Econometrica*, 69(4):959–993, 2001.

[56] Bjørn Eraker. Mcmc analysis of diffusion models with application to finance. *Journal of Business and Economic Statistics*, 19(2):177–191, 2001.

[57] G.O. Roberts and O. Stramer. On inference for partially observed nonlinear diffusion models using the metropolis–hastings algorithm. *Biometrika*, 88(3):603, 2001.

[58] Kalogeropoulos. Bayesean inference for multidimensional diffusion processes. *PhD thesis*, 2006.

[59] Y. Aït-Sahalia. Closed-form likelihood expansions for multivariate diffusions. *The Annals of Statistics*, 36(2):906–937, 2008.

[60] Andrew Golightly and Darren J Wilkinson. Bayesian inference for nonlinear multivariate diffusion models observed with error. *Computational Statistics & Data Analysis*, 52(3):1674–1693, 2008.

[61] Alexandros Beskos and Gareth O Roberts. Exact simulation of diffusions. *The Annals of Applied Probability*, 15(4):2422–2444, 2005.

[62] A. Beskos, O. Papaspiliopoulos, and G.O. Roberts. Retrospective exact simulation of diffusion sample paths with applications. *Bernoulli*, 12(6):1077, 2006.

[63] A. Beskos, O. Papaspiliopoulos, G.O. Roberts, and P. Fearnhead. Exact and computationally efficient likelihood-based estimation for discretely observed diffusion processes (with discussion). *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68:333–382, 2006.

[64] A. Beskos, G. Roberts, A.M. Stuart, and J. Voss. An mcmc method for diffusion bridges. *Stochastics and Dynamics*, 8:319–350, 2008.

[65] Christophe Andrieu, Arnaud Doucet, and Roman Holenstein. Particle Markov Chain Monte Carlo methods. *Journal of the Royal Statistical Society*, 72:1–33, 2010.

[66] Christophe Andrieu and Gareth O Roberts. The pseudo-marginal approach for efficient monte carlo computations. *The Annals of Statistics*, 37(2):697–725, 2009.

[67] G.B. Durham and A.R. Gallant. Numerical techniques for maximum likelihood estimation of continuous-time diffusion processes (with comments). *Journal of Business and Economic Statistics*, 20:297–338, 2002.

[68] C. Archambeau, D. Cornford, M. Opper, and J. Shawe-Taylor. Gaussian process approximations of stochastic differential equations. *JMLR: Workshop and Conference Proceedings*, 1:1–16, 2007.

[69] Y. Shen, C. Archambeau, D. Cornford, M. Opper, J. Shawe-Taylor, and R. Barillec. A comparison of variational and Markov chain Monte Carlo methods for inference in partially observed stochastic dynamic systems. *Journal of Signal Processing Systems*, 61(1):51–59, 2010.

[70] Y. Aït-Sahalia. Maximum likelihood estimation of discretely sampled diffusions: a closed-form approximation approach. *Econometrica*, 70:223–262, 2002.

[71] Y Aıt-Sahalia. Likelihood inference for diffusions: a survey. *Frontiers in Statistics: in Honor of Peter J. Bickels 65th Birthday*, pages 369–407, 2006.

[72] A. Beskos, O. Papaspiliopoulos, and G.O. Roberts. Monte-Carlo maximum likelihood estimation for discretely observed diffusion processes. *Annals of Statistics*, 37:223–245, 2009.

[73] W. Luo. *Wiener chaos expansion and numerical solutions of stochastic partial differential equations*. PhD thesis, California Institute of Technology, 2006.

[74] E. Wong and M. Zakai. On the convergence of ordinary integrals to stochastic integrals. *The Annals of Mathematical Statistics*, pages 1560–1564, 1965.

[75] EJ McShane. Stochastic differential equations and models of random processes. In *Proceedings of the Sixth Berkeley Symposium on Mathematical Statistics and Probability (Univ. California, Berkeley, Calif., 1970/1971)*, volume 3, pages 263–294, 1972.

[76] Halim Doss. Liens entre équations différentielles stochastiques et ordinaires. In *Annales de l'institut Henri Poincaré (B) Probabilités et Statistiques*, volume 13, pages 99–125. Gauthier-Villars, 1977.

[77] Héctor J Sussmann. On the gap between deterministic and stochastic ordinary differential equations. *The Annals of Probability*, 6(1):19–41, 1978.

[78] H. Singer. Parameter estimation of nonlinear stochastic differential equations: simulated maximum likelihood versus extended Kalman filter and Itô-Taylor expansion. *Journal of Computational and Graphical Statistics*, 11(4):972–995, 2002.

[79] M. Opper, A. Ruttor, and G. Sanguinetti. Approximate inference in continuous time Gaussian-jump processes. *Advances in Neural Information Processing Systems*, 23:1831–1839, 2010.

[80] I. Arasaratnam, S. Haykin, and T.R. Hurd. Cubature Kalman filtering for continuous-discrete systems: Theory and simulations. *Signal Processing, IEEE Transactions on*, 58(10):4977–4993, 2010.

[81] Endre Süli and David F Mayers. *An introduction to numerical analysis*. Cambridge university press, 2003.

[82] S. Chib, M.K. Pitt, and N. Shepard. Likelihood-based inference for diffusion models. *Working Paper*, 2004. http://www.nuff.ox.ac.uk/economics/papers/2004/w20/chibpittshephard.pdf.

[83] D. Rimmer, A. Doucet, and W.J. Fitzgerald. Particle filters for stochastic differential equations of nonlinear diffusions. Technical report, Cambridge University Engineering Department, 2005.

[84] H. Singer. Nonlinear continuous time modeling approaches in panel research. *Statistica Neerlandica*, 62(1):29–57, 2008.

[85] S. Corlay and P. Gilles. Functional quantization based stratified sampling methods. *Arxiv preprint Arxiv:1008.4441*, 2010.

[86] A.F. Bastani and S.M. Hosseini. A new adaptive Runge-Kutta method for stochastic differential equations. *Journal of Computational and Applied Mathematics*, 206:631–644, 2007.

[87] N.G. van Kampen. *Stochastic processes in physics and chemistry*. North holland, 2007.

[88] Abigail A Flower, J Randall Moorman, Douglas E Lake, and John B Delos. Periodic heart rate decelerations in premature infants. *Experimental Biology and Medicine*, 235(4):531–538, 2010.

[89] S. Särkkä and T. Sottinen. Application of Girsanov theorem to particle filtering of discretely observed continuous-time non-linear systems. *Bayesian Analysis*, 3(3):555–584, 2008.

[90] L.M. Murray and A.J. Storkey. Particle smoothing in continuous time: A fast approach via density estimation. *IEEE Transactions on Signal Processing*, 59:1017–1026, 2011.

[91] Jayesh H Kotecha and Petar M Djuric. Gaussian particle filtering. *Signal Processing, IEEE Transactions on*, 51(10):2592–2601, 2003.

[92] Karl J Friston. Variational filtering. *NeuroImage*, 41(3):747–766, 2008.

[93] Fred Daum and Jim Huang. Particle flow for nonlinear filters with log-homotopy. In *SPIE Defense and Security Symposium*, volume 6969, 2008.

[94] Bhashyam Balaji. Continuous-discrete path integral filtering. *Entropy*, 11(3):402–430, 2009.

[95] H. Kushner. Approximations to optimal nonlinear filters. *Automatic Control, IEEE Transactions on*, 12(5):546–556, 1967.

[96] K. Ito and K. Xiong. Gaussian filters for nonlinear filtering problems. *Automatic Control, IEEE Transactions on*, 45(5):910–927, 2000.

[97] Simon Julier, Jeffrey Uhlmann, and Hugh F Durrant-Whyte. A new method for the nonlinear transformation of means and covariances in filters and estimators. *Automatic Control, IEEE Transactions on*, 45(3):477–482, 2000.

[98] I. Arasaratnam and S. Haykin. Cubature Kalman filters. *Automatic Control, IEEE Transactions on*, 54(6):1254–1269, 2009.

[99] S. Särkkä and A. Solin. On continuous-discrete cubature Kalman filtering. *Proc. SYSID 2012*, pages 1210–1215.

[100] P. Friz and N. Victoir. Differential equations driven by Gaussian signals. In *Annales de l'Institut Henri Poincaré, Probabilités et Statistiques*, volume 46, pages 369–413. Institut Henri Poincaré, 2010.

[101] S. Särkkä and J Hartikainen. On gaussian optimal smoothing of non-linear state space models. *Automatic Control, IEEE Transactions on*, 55(8):1938–1941, 2010.

[102] Ryan Turner and Carl Edward Rasmussen. Model based learning of sigma points in unscented Kalman filtering. *Neurocomputing*, 80:47–53, 2012.

[103] Fredrik Gustafsson and Gustaf Hendeby. Some relations between extended and unscented Kalman filters. *Signal Processing, IEEE Transactions on*, 60(2):545–555, 2012.

[104] Eric A Wan and Rudolph Van Der Merwe. The unscented Kalman filter for nonlinear estimation. In *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000*, pages 153–158. IEEE, 2000.

[105] Vasileios Maroulas and Panos Stinis. A Girsanov Monte Carlo approach to particle filtering for multi-target tracking. Technical report, University of Minnesota, 2010.

[106] I. Gyongy, A. Shmatkov, et al. Rate of convergence of Wong-Zakai approximations for stochastic partial differential equations. *Applied Mathematics and Optimization*, 54(3):341–341, 2006.

[107] N. Ikeda and S. Watanabe. Stochastic differential equations and diffusion processes. 1989.