The Iterated Auxiliary Particle Filter and Applications to State Space Models and Diffusion Processes

Pieralberto Guarniero

Thesis submitted to the University of Warwick
for the degree of
Doctor of Philosophy
Department of Statistics

August 2017

Abstract

The novel research work presented in this thesis consists of an offline, iterated particle filter to facilitate statistical inference in general state space hidden Markov models. Given a model and a sequence of observations, the associated marginal likelihood L is central to likelihood-based inference for unknown statistical parameters. We define a class of "twisted" models: each member is specified by a sequence of positive functions $\boldsymbol{\psi}$ and has an associated $\boldsymbol{\psi}$-auxiliary particle filter that provides unbiased estimates of L. We identify a sequence $\boldsymbol{\psi}^{\boldsymbol{*}}$ that is optimal in the sense that the $\boldsymbol{\psi}^{\boldsymbol{*}}$-auxiliary particle filter's estimate of L has zero variance. In practical applications, $\boldsymbol{\psi}^{\boldsymbol{*}}$ is unknown so the $\boldsymbol{\psi}^{\boldsymbol{*}}$- auxiliary particle filter cannot straightforwardly be implemented. We use an iterative scheme to approximate $\boldsymbol{\psi}^*$, and demonstrate empirically that the resulting iterated auxiliary particle filter significantly outperforms the most popular competitors in some challenging settings. Applications include parameter estimation using a particle Markov chain Monte Carlo algorithm. An adaptation of the iAPF for statistical inference in the context of diffusion processes along with a number of examples and applications in this setting is provided.

Declaration

This thesis is the result of my own work and research, except where otherwise indicated. The content of Part II and Part III includes, but is not limited to, the work contained in the published journal article

Guarniero, P., Johansen, A. M. & Lee, A. (2016), 'The iterated auxiliary particle filter', Journal of the American Statistical Association (just accepted)

This thesis has not been submitted for examination to any other institution than the University of Warwick.

Pieralberto Guarniero

Acknowledgments

My supervisors Adam Johansen and Anthony Lee have been my invaluable guiding stars throughout the course of my PhD. They are extremely talented and humble. They are creative and pragmatic, providing me with countless clever and viable ideas. They have been demanding and ambitious when needed and understanding and supportive when I was falling behind. I owe so much to them and I truly believe they have been the best guidance I could have desired.

Gareth Roberts, Paul Jenkins and Mark Girolami, as part of my PhD internal panel, have provided me with crucial feedback and cunning ideas. The constructive discussion during the viva with my external examiners Chris Sherlock and Petros Dellaportas, along with their corrections and notes, have contributed to significantly improve the final version of this thesis both in content and in form. It has been an honour to see all these senior academics showing genuine interest for my research and every examination encounter up to and including the viva has been an important and fruitful occasion to hear their sharp insights into my work.

Wolfgang Runggaldier and Paolo Dai Pra, my undergraduate and postgraduate supervisors, have encouraged me in undertaking the doctorate path at Warwick. Their support filled me with enthusiasm and determination which are essential resources especially at the beginning of a PhD course.

The support and cheer I have received from my family is unmatched. They are at least as happy as I am for the achievement of writing this PhD thesis.

Sometimes producing research and writing a PhD thesis might feel like a lonely process. Throughout these years it was sufficient to raise my head and look at the good friends and housemates surrounding me to realise this is not the case. Among these angels I would like to mention some who have played an important part in my academic life, with their minds and with their hearts: Christiane Olivia, Silvia Calderazzo, Alejandra Avalos Pacheco, Felipe Medina Aguayo, Dialid Santiago, Thomas Honnor, Elke Thönnes, Jon Warren, Elia Gironacci, Javier Fernandez, Matt Teft, Riccardo Sambo, Alex Nardi.

Contents

Chapter 1

Introduction

Context

Since what is considered to be their first instance in Gordon et al. (1993), sequential Monte Carlo methods (SMC), or particle filters, have become extremely popular and their application, initially limited to tracking and vision problems, has spread to many areas such as finance and risk analysis, engineering and robotics, biology and molecular chemistry and other fields; see Doucet et al. (2001) for a comprehensive review of the literature. Their most prominent application is for the solution of optimal estimation problems in non-linear non-Gaussian Bayesian dynamical systems. The common denominator of such models is that a latent stochastic process, for instance a set of kinetic characteristics of a moving target in a tracking application or the common underlying volatility of a set of financial securities in a financial problem, is known only through a sequence of noisy measurements, namely the observation process. The main advantage of SMC methods is that in general they do not require the imposition of any form of linearisation or crude approximation on the system dynamic and that in principle their implementation is straightforward provided some essential features of the inherent dynamic systems are available in a probabilistic form. Their main drawback is their computational cost.

When the dynamic system under investigation possesses a strong memory structure, future information can help sharpen the inference about the current state of the latent process, or equivalently lighten the computational burden of a SMC algorithm. Look-ahead methods constitute a subset of SMC methods that follows the basic principle of using future information for designing particle filters. Literature on look-ahead methods is very prolific (Pitt & Shephard (1999), Doucet et al. (2006), Lin et al. (2013), Zhang & Liu (2002)) and some outstanding results have been achieved. However, the performance of most look-ahead schemes depends critically on some form of algorithm tailoring which is specific to the application and non trivial.

The main focus of this thesis is a novel look-ahead scheme, namely the iterated auxiliary particle filter (iAPF) and its applications to state space models and diffusion processes. Simulation results for these applications suggest that in some interesting scenarios the iAPF can lead to significant improvements in the precision of statistical estimates with respect to standard non look-ahead schemes, thanks to an effective exploitation of future information. Its main advantage with respect to other existing look-ahead schemes is that it consists of an automated iterative exploration procedure which directly applies in a rather general setting without any tailoring apart from the choice of some precision parameters for which general guidelines are provided.

Outline

This thesis is divided into four parts, each part consisting of a number of short chapters. Novel methodology, examples and applications compose the last three parts. The content of Part II and Part III includes, but is not limited to, the work contained in the published journal article

Guarniero, P., Johansen, A. M. & Lee, A. (2016), 'The iterated auxiliary particle filter', Journal of the American Statistical Association (just accepted).

Part I In Chapter 2 we define hidden Markov models (HMM) which are the dynamic Bayesian systems with a particular independence structure under investigation. Statistical inference for hidden Markov models is the original motivating application for SMC methodologies and still remains one of their most prominent applications. In Chapter 3 we present SMC methods, which involve the simulation over time of an artificial particle system, under a rather general framework.

Here we briefly introduce the basics of Monte Carlo methods and Importance Sampling (IS). In SMC methods, IS is performed in an iterative fashion and the feature of particle resampling is introduced to address the degeneracy of importance weights problem. A basic algorithm that encompasses a wide class of SMC techniques is provided along with an essential central limit theorem. In Chapter 4 we briefly review look-ahead methods, a subset of SMC methods that aims at producing high precision SMC schemes by using future information for designing the behaviour of the SMC particle system. The iAPF algorithm which is the focus of this thesis belongs to this class of methods. We also provide a brief description of particle marginal Metropolis Hasting (PMMH), a powerful parameter estimation method that combines standard Markov chain Monte Carlo and SMC schemes, as we use it extensively in our simulations.

Part 2 In Chapter 5 we define a class of particle filters parametrised by an appropriate sequence of functionals. Our interest in this family of schemes is motivated by the fact that our main statistical quantity of interest is common to the entire family. Many popular look-ahead schemes can be retrieved from this class. In particular we provide an optimal sequence of functionals which leads to a zero variance look-ahead scheme, in an appropriate sense. Such an idealised particle filter gives motivational ground for the iAPF algorithm of Chapter 6. The iAPF works in an iterative fashion, each iteration consisting of a particle filter which belongs to the aformentioned family. Through a completely automated procedure, at each iteration the algorithm exploits the output of the previous iteration to define a new sequence of functionals as close as possible, in an appropriate sense, to the optimal sequence, possibly leading to a particle filter with enhanced accuracy. A stopping rule based on the fluctuation of subsequent estimates is given. In Chapter 7 some important implementation details are provided. In particular we discuss two possible approaches to define the parametrising sequence of functionals: a kernel density estimate approach and a parametric approach.

Part 3 We perform simulations for different HMMs to showcase the iAPF algorithm. In Chapter 8 we consider some linear Gaussian models for which the quantities under investigation are known in an analytic form, facilitating comparisons with the Bootstrap particle filter (BPF) and the fully adapted particle filter, two SMC competitors of the iAPF. In these examples the benefits of our approach are particularly significant in the case of extreme observations and high dimensions. In Chapter 9 we use the iAPF for statistical inference for stochastic volatility models, using some real world data. First we consider the parameter estimation problem for a univariate stochastic volatility model based on a sequence of pound/dollar exchange rate time series. In this application the iAPF performs only slightly better than the BPF, although results indicate that the iAPF estimates are significantly less variable across the parameter range than their BPF counterparts, and may therefore be more suitable in simulated maximum likelihood approximations. With the second most challenging application we look at a multivariate stochastic volatility model, and the parameter estimation problem given monthly returns for the exchange rate with respect to the US dollar of a range of 20 different international currencies. In this scenario because of the relatively high dimensionality of the state space and of the 79-dimensional parameter space the BPF systematically fails to provide reasonable estimates in a feasible computational time. The iAPF instead manages to produce reasonable and potentially interesting results in this highly-challenging context.

Part 4 In this last part of the thesis we explore the potential of a version of the iterated auxiliary particle filter to make inference on a diffusion process setting. In the context of diffusion processes it is often convenient to introduce a discrete-time approximation that converges in an appropriate sense to the continuous-path diffusion as the discretisation step tends to zero. When the continuous process is partially or perfectly observed at discrete time steps, we can interpret its discrete approximation as a state space model, for which statistical inference through the iAPF becomes possible. In Chapter 10 we consider a general class of diffusion processes and we show how the problem of estimating transition densities of a continuous diffusion in this class through its Euler-Maruyama approximation can be reformulated as an SMC estimation problem. In Chapter 11 we describe two modifications to the iAPF that can significantly enhance its performance in this setting, especially when a fine Euler–Maruyama approximation is required. In Chapter 12 we present simulation results for different diffusion processes to assess the performance of the iAPF with respect to the BPF for the problem of estimating transition densities and sampling from diffusion bridges. We also review briefly the major competing methods for estimating transition densities in the context of diffusion processes and perform some simulations to compare the iAPF with the modified diffusion bridge approach.

Part I

Look-Ahead Methods for Sequential Monte Carlo

Chapter 2

Hidden Markov Models

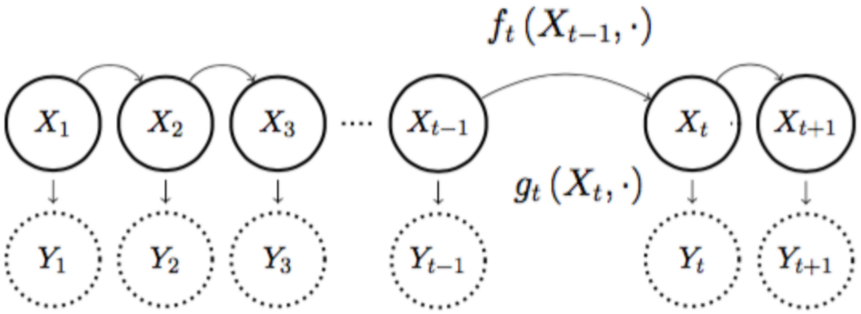
2.1 Definition

A hidden Markov model (HMM) is a bivariate Markov chain $(X_t, Y_t)_{t\geq 1}$ with a particular conditional independence structure, where only the process $(Y_t)_{t\geq 1}$ is observed (Künsch (2000)). The hidden process $(X_t)_{t\geq 1}$ is itself a Markov chain evolving on the state space X and the observation process $(Y_t)_{t\geq 1}$ is defined on the observation space Y. Each Y_t is conditionally independent on all the other random variables (i.e. $\{X_i, Y_j : i, j \neq t\}$) given X_t. We fix a final time $T \in \mathbb{N}$ and we have

$$\begin{aligned} X_1 &\sim \mu\left(\cdot\right) & \\ X_t \mid \{X_{t-1} = x_{t-1}\} &\sim f\left(x_{t-1}, \cdot\right) & \text{for } t \in 2:T \\ Y_t \mid \{X_t = x_t\} &\sim g\left(x_t, \cdot\right) & \text{for } t \in 1:T \end{aligned}$$

where $\mu : \mathsf{X} \to \mathbb{R}_+$ is the initial probability density function, $f : \mathsf{X} \times \mathsf{X} \to \mathbb{R}_+$ a transition density and $g : \mathsf{X} \times \mathsf{Y} \to \mathbb{R}_+$ an observation density. We call such a construction a HMM (μ, f, g). In the following we treat the case where $(\mathsf{X}, \mathcal{B}(\mathsf{X})) = \left(\mathbb{R}^d, \mathcal{B}\left(\mathbb{R}^d\right)\right)$ and $(\mathsf{Y}, \mathcal{B}(\mathsf{Y})) = \left(\mathbb{R}^{d'}, \mathcal{B}\left(\mathbb{R}^{d'}\right)\right)$ that is when X and Y are real coordinate measurable spaces with the associated Borel $\sigma-$algebra, generated by the open sets. All the densities are with respect to the common Lebesgue dominating measure. We use the notation $k : T := \{k, k+1, \ldots, T\}$ for all $k \leq T$ and specify when the order of the elements matters only when it is not clear from the context. The generalisation to time-inhomogeneous transition and observation distributions and time-dependent state and observation spaces is straightforward. We choose this setting for a clearer exposition.

A general HMM can be illustrated with the diagram below.



Statistical inference is often conducted upon the basis of a realization $y_{1:T}$ of $Y_{1:T}$, which we will consider to be fixed. A HMM defines a Bayesian model where the prior density of the process of interest $(X_t)_{t\in 1:T}$ is given by

$$p(x_{1:T}) = \mu(x_1) \prod_{t=2}^{T} f(x_{t-1}, x_t)$$

and the likelihood associated with a realisation of the observations process $Y_{1:T} = y_{1:T}$ is given by

$$p(y_{1:T} \mid x_{1:T}) = \prod_{t=1}^{T} g(x_t, y_t).$$

In this context the posterior density which is proportional to the product of the prior density $p(x_{1:T})$ and the likelihood function $p(y_{1:T} \mid x_{1:T})$ takes the form

$$p(x_{1:T} \mid y_{1:T}) = \frac{p(x_{1:T}, y_{1:T})}{p(y_{1:T})},$$

where $p(x_{1:T}, y_{1:T}) = p(x_{1:T})\, p(y_{1:T} \mid x_{1:T})$ and $p(y_{1:T}) = \int_{\mathsf{X}^T} p(x_{1:T}, y_{1:Y})\, dx_{1:T}$.

We denote by $\mathbb{P}$ the law of the bivariate Markov chain $(X_t, Y_t)_{t\in 1:T}$

$$\mathbb{P}(X_{1:T} \in A, Y_{1:T} \in B) := \int_{A\times B} \mu(x_1)\, g(x_1, y_1) \prod_{t=2}^{T} f(x_{t-1}, x_t)\, g(x_t, y_t)\, dx_{1:T} dy_{1:T}$$

for any Borel measurable set $A \subseteq \mathsf{X}^T$ and $B \subseteq \mathsf{Y}^T$, $dx_{1:T} = dx_1 dx_2 \cdots dx_T$. Letting $\mathbb{E}$ denote expectations w.r.t. $\mathbb{P}$, for any sequence of observations $y_{1:T}$ the conditional path probability measure is given by

$$\mathbb{P}\left(X_{1:T} \in A \mid Y_{1:T} = y_{1:T}\right) = \frac{\mathbb{E}\left[\mathbb{I}_A\left(X_{1:T}\right) \prod_{t=1}^{T} g\left(X_t, y_t\right)\right]}{\mathbb{E}\left[\prod_{t=1}^{T} g\left(X_t, y_t\right)\right]}$$

for any Borel measurable $A \subseteq \mathsf{X}^T$. In some applications the objects of inference are the conditional distributions $X_T \mid Y_{1:T}$ and $X_{1:T} \mid Y_{1:T}$ which are called respectively the filtering distribution and the smoothing distribution. Our main statistical quantity of interest is $L := \mathbb{E}\left[\prod_{t=1}^{T} g\left(X_t, y_t\right)\right]$, the marginal likelihood associated with $y_{1:T}$. In most of our applications and examples, an unknown statistical parameter $\theta \in \Theta$ governs μ, f and g, and in this setting the map $\theta \mapsto \mathrm{L}(\theta)$ is the likelihood function. Other possible quantities of interest are the so-called smoothing expectations, i.e. $\mathbb{E}\left[\phi\left(X_{1:T}\right) \mid Y_{1:T} = y_{1:T}\right]$ for some bounded continuous function $\phi : \mathsf{X}^T \longrightarrow \mathbb{R}$.

2.2 Examples

2.2.1 The linear Gaussian model

The linear Gaussian model is a HMM with state space $\mathsf{X} = \mathbb{R}^d$ and observation space $\mathsf{Y} = \mathbb{R}^{\mathsf{d}'}$ defined by the following initial, transition and observation Gaussian densities:

$$\mu(\cdot) = \mathcal{N}(\cdot; m, \Sigma)$$

$$f(x, \cdot) = \mathcal{N}(\cdot; Ax, B)$$

$$g(x, \cdot) = \mathcal{N}(\cdot; Cx, D)$$

where $m \in \mathbb{R}^d$, $\Sigma, A, B \in \mathbb{R}^{d \times d}$, $C \in \mathbb{R}^{d \times d'}$ and $D \in \mathbb{R}^{d' \times d'}$. $\mathcal{N}(\cdot; m, \Sigma)$ denotes a possibly multidimensional Gaussian distribution with vector mean m and covariance matrix Σ. We take into consideration this model because it is possible to derive analytic expressions for its normalising constant L, filtering distribution and smoothing distributions through a recursive algorithm called the Kalman filter (see, for example, Ghahramani (1998)). For the Linear

Gaussian model the predictive distribution $X_t \mid Y_{1:t-1}$ and the filtering distribution $X_t \mid Y_{1:t}$ are Gaussian distributions for all $t \in 1 : T$. The Kalman filter is based on a set of equations that determine the parameters of such Gaussian distributions recursively. Let $m_{t|t-1}, m_{t|t} \in \mathbb{R}^d$ and $\Sigma_{t|t-1}, \Sigma_{t|t} \in \mathbb{R}^{d\times d}$ be the means and variance/covariance matrices of the predictive and filtering distribution at time t respectively, so that $X_t \mid Y_{1:t-1} \sim \mathcal{N}\left(m_{t|t-1}, \Sigma_{t|t-1}\right)$ and $X_t \mid Y_{1:t} \sim \mathcal{N}\left(m_{t|t}, \Sigma_{t|t}\right)$ for $t \in 1 : T$, with the convention $X_1 \mid Y_{1:0} = X_1$. The predict equations allow one to determine the parameters of the predictive distribution $X_t \mid Y_{1:t-1}$ given the distribution $X_{t-1} \mid Y_{1:t-1}$. The update equations define the parameters of the filtering distribution $X_t \mid Y_{1:t}$ at time t given the predictive distribution $X_t \mid Y_{1:t-1}$. Starting from the predictive distribution $X_1 \sim \mathcal{N}\left(m_{1|0}, \Sigma_{1|0}\right)$ at time $t = 0$ which trivially corresponds to the distribution with density $\mu(\cdot) \sim \mathcal{N}(\cdot; m, \Sigma)$, the Kalman filter alternates update and predict steps determining the filtering distribution at each time t up to T. For a fixed sequence of observations $y_{1:T}$, and given the parameters $m_{t|t-1}$, $\Sigma_{t|t-1}$ of the predictive distribution at time $t \in 1 : T$ the update step consists of the following set of equations

$$\begin{aligned}
S_t &= C\Sigma_{t|t-1}C^t + D \\
G_t &= \Sigma_{t|t-1}C^t S_t^{-1} \\
m_{t|t} &= m_{t|t-1} + G_t\left(y_t - \hat{y}_t\right) \\
\Sigma_{t|t} &= \Sigma_{t|t-1} - G_t S_t G_t^t
\end{aligned}$$

where

$$\hat{y}_t = Cm_{t|t-1}$$

is the observation prediction, $G_t \in \mathbb{R}^{d\times d'}$ is the Kalman gain matrix and $S_t \in \mathbb{R}^{d'\times d'}$ is the observation prediction covariance at time t. The prediction step consists of the equations

$$\begin{aligned}
m_{t+1|t} &= Am_{t|t} \\
\Sigma_{t+1|t} &= A\Sigma_{t|t}A^t + B
\end{aligned}$$

where the parameters $m_{t|t}$ and $\Sigma_{t|t}$ are obtained from the update step at time t.

From the Kalman equations it is also possible to derive the normalising constants Z_t for every $t \in 1:T$. In particular we have that

$$p\left(y_t \mid y_{1:t-1}\right) = \mathcal{N}\left(y_t; \hat{y}_t, S_t\right)$$

for all $t \in 1 : T$ and therefore $Z_t = p\left(y_{1:T}\right)$ is straightforwardly derived as $p\left(y_{1:T}\right) = p\left(y_1\right) \prod_{t=2}^{T} p\left(y_t \mid y_{1:t-1}\right)$. Quantities such as $p\left(y_{t+1:T} \mid x_t\right)$, which appear in the idealised version of the algorithm we present in Part II of the thesis, are also easily derived through the Kalman filter. It is sufficient to notice that $p\left(y_{t+1:T} \mid x_t\right)$ corresponds to the marginal likelihood $p\left(y'_{1:T-t}\right)$ relative to the sequence of observations $y'_{1:T-t} = y_{t+1:T}$ from the linear Gaussian HMM with initial distribution $\mu\left(\cdot\right) \sim \mathcal{N}\left(\cdot; Ax_t, B\right)$, transition density $f\left(x, \cdot\right) = \mathcal{N}\left(\cdot; Ax, B\right)$ and observation density $g\left(x, \cdot\right) = \mathcal{N}\left(\cdot; Cx, D\right)$. The fact that we can obtain analytic expressions for these distributions and quantities facilitates the comparison between the algorithms presented in the following chapters.

2.2.2 Stochastic volatility models

Stochastic volatility models have been widely used in various areas of economics and mathematical finance. They describe the dynamic of a set of financial securities whose volatilities are themselves random processes. Some classical models such as the Black–Scholes model make the simplifying assumption that the underlying volatility of the market is constant. Under this assumption many features of a realistic financial market dynamic cannot be modelled. Some of these features, such as smile and skew of the implied stochastic volatility surface, can instead be described by assuming that the volatility process underlying the modelled financial securities is a stochastic process rather than a constant.

A classical example of a simple stochastic volatility model is defined by $\mu\left(\cdot\right) = \mathcal{N}\left(\cdot; 0, \sigma^2/\left(1-\alpha\right)^2\right)$, $f\left(x, \cdot\right) = \mathcal{N}\left(\cdot; \alpha x, \sigma^2\right)$ and $g\left(x, \cdot\right) = \mathcal{N}\left(\cdot; 0, \beta^2 \exp\left(x\right)\right)$ where $\alpha \in (0, 1)$, $\beta > 0$ and $\sigma^2 > 0$ are statistical parameters (see, e.g. Kim et al. (1998)).

In Part III we also consider a multivariate stochastic volatility model that can also be found in Chib et al. (2009, Section 2). The model is defined for

$\mathsf{X} = \mathbb{R}^d$ by $\mu(\cdot) = \mathcal{N}(\cdot; m, U_\star)$, $f(x, \cdot) = \mathcal{N}(\cdot; m + \mathrm{diag}(\phi)\,(x - m)\,, U)$ and $g(x, \cdot) = \mathcal{N}\left(\cdot; 0, \exp\left(\mathrm{diag}\left(x\right)\right)\right)$, where $m, \phi \in \mathbb{R}^d$ and the covariance matrices $U, U_\star \in \mathbb{R}^{d \times d}$ are statistical parameters.

Chapter 3

Sequential Monte Carlo Methods

The Sequential Monte Carlo (SMC) methodology involves the simulation over time of an artificial particle system $(\xi_t^i : t \in \{1, \ldots, T\}, i \in \{1, \ldots, N\})$ from which we can derive weighted samples from a sequence of target probability densities $(\pi_t)_{t \in 1:T}$ of increasing dimension, where π_t is defined on the product space X^t. We are normally able to evaluate π_t only up to an unknown normalising constant that is we can evaluate pointwise the function γ_t where

$$\pi_t(x_{1:t}) = \frac{\gamma_t(x_{1:t})}{Z_t}$$

for all $t \in 1 : T$. Given a density function π_t defined on the measurable space $(\mathsf{X}^{\mathsf{t}}, \mathcal{B}(\mathsf{X}^{\mathsf{t}}))$ we say that a test function $\phi_t : \mathsf{X}^t \longrightarrow \mathbb{R}$ is π_t-integrable if the integral $\int_{\mathsf{X}^t} \phi_t(x_{1:t}) \pi_t(x_{1:t}) dx_{1:t}$ exists and it is finite and we omit the π_t when it is clear from the context. Depending on the application, the main focus can be either to produce samples from the distribution densities $(\pi_t)_{t \in 1:T}$, or to estimate the normalising constant $Z_t = \int \gamma_t(x_{1:t}) dx_{1:t}$, or to evaluate some quantity of the form $\pi_t(\phi_t) := \int_{\mathsf{X}^t} \phi_t(x_{1:t}) \pi_t(x_{1:t}) dx_{1:t}$ for some $t \in 1 : T$ and some integrable test function $\phi_t : \mathsf{X}^t \longrightarrow \mathbb{R}$. In any case SMC methods produce a sequential approximation of the aforementioned quantities, for example first an estimate of Z_1 then, after appropriately propagating the particle system, an estimate of Z_2 and so on.

SMC methods were originally applied to statistical inference for hidden Markov models (HMMs) by Gordon et al. (1993) under the name of particle filters, and this setting remains an important application. In this context we could have $\gamma_t(x_{1:t}) = p(x_{1:t}, y_{1:t})$ and thus the target densities $\pi_t(x_{1:t}) = p(x_{1:t} \mid y_{1:t})$ assume the form of conditional densities, and the normalising constants $Z_t = p(y_{1:t})$ are interpreted as marginal likelihoods associated with the sequence of observations $y_{1:t}$. This is just a particular choice of target distributions. With our approach described in Part II and involving twisted models, for example, the final target distribution $\pi_T(x_{1:T}) = p(x_{1:T} \mid y_{1:T})$ coincides with the full conditional distribution given the observation sequence $y_{1:T}$, but the intermediate distributions π_t for $t \in 1 : T-1$ take the form of twisted distributions and thus in general $\pi_t(x_{1:t}) \neq p(x_{1:t} \mid y_{1:t})$.

Despite the broad class of applications, it is possible to present SMC methods under a general framework and notation, and with a basic algorithm we can encompass a wide class of advanced SMC techniques. In the rest of this paragraph we present such general framework following Doucet & Johansen (2011). First we briefly introduce the basics of Monte Carlo methods and Importance Sampling (IS). Then we see how, when an appropriate decomposition of the proposal distribution is possible, IS can be applied in a sequential fashion: in this case it takes the name of Sequential Importance Sampling (SIS). SMC is a modification of SIS where particle resampling is introduced to address the degeneracy of importance weights problem.

3.1 Basics of Monte Carlo methods

Monte Carlo methods are a broad class of computational algorithms that rely on repeated random sampling with the aim of approximating measures and, by extension, integrals. We consider the problem of approximating a generic probability density $\pi_t(x_{1:t})$ defined on the state space X^t. This description in terms of t-dimensional distributions is functional to our specific HMM setting. If it is possible to repeatedly sample from such distribution, we can construct a Monte Carlo approximation of the distribution with density π_t as the empirical probability measure

$$\pi_t^N(dx_{1:t}) = \frac{1}{N}\sum_{i=1}^{N} \delta_{\xi_{1:t}^i}(dx_{1:t})$$

where $\xi_{1:t}^{i} \sim \pi_t$ are independent and identically distributed for $i \in 1:N$, $N \geq 1$ and δ_ξ is the Dirac delta (i.e. a point mass) located at $\xi \in \mathsf{X}^t$. For any integrable test function $\phi_t : \mathsf{X}^t \longrightarrow \mathbb{R}$ we can approximate the integral $\pi_t(\phi_t)$ by the Monte Carlo approximation $\pi_t^N(\phi_t) := \frac{1}{N}\sum_{i=1}^{N} \phi_t(\xi_{1:t}^{i})$. It is easily checked that $\pi_t^N(\phi)$ is an unbiased estimator of $\pi_t(\phi)$ as

$$\mathbb{E}\left[\pi_t^N(\phi_t)\right] \;=\; \mathbb{E}\left[\phi_t\left(\xi_{1:t}^{1}\right)\right]$$

because $\xi_{1:t}^{i}$ are independent and identically distributed according to π_t, and the term on the right hand side is equal to $\pi_t(\phi_t)$ by definition. Furthermore the law of large numbers ensures that $\pi_t^N(\phi_t) \xrightarrow{N\longrightarrow\infty} \pi_t(\phi_t)$ almost surely for any integrable ϕ. The main advantage of the Monte Carlo estimator $\pi_t^N(\phi_t)$ is that its variance

$$\mathrm{Var}\left(\pi_t^N(\phi_t)\right) = \frac{1}{N}\left(\pi_t\left(\phi_t^2\right) - \left(\pi_t(\phi_t)\right)^2\right)$$

decreases as N increases at a rate that does not depend on the dimensionality of the state space X^t while other standard numerical approximations such as quadrature rules have a convergence rate that degrades rapidly with higher dimension: this is known as the curse of dimensionality. In most scenarios and applications of interest sampling directly from the target distribution π_t is impossible or computationally unfeasible, a problem that can be addressed with the introduction of an alternative proposal distribution q_t.

3.2 Importance sampling

Importance Sampling relies on the introduction of a proposal density q_t from which we can easily sample and such that $\mathrm{supp}(q_t) \supseteq \mathrm{supp}(\pi_t)$, where $\mathrm{supp}(f) := \{x \in \mathsf{X} : f(x) > 0\}$ for any non negative real function f on X. The identity

$$\pi_t(\phi_t) = \int_{\mathsf{X}^{\mathsf{t}}} \phi_t(x_{1:t})\,\pi_t(x_{1:t})\,dx_{1:t} = \int_{\mathrm{supp}(q_t)} \phi_t(x_{1:t})\,\frac{\pi_t(x_{1:t})}{q_t(x_{1:t})} q_t(x_{1:t})\,dx_{1:t}$$

suggests an alternative procedure to derive estimates of the quantity $\pi_t(\phi_t)$ for an integrable test function ϕ_t. We draw N independent identically distributed samples $\xi^i_{1:t} \sim q_t$ and we obtain the approximation

$$\pi^N_t\left(\phi_t\right) = \frac{1}{N}\sum_{i=1}^{N}\phi_t\left(\xi^i_{1:t}\right)\frac{\pi_t\left(\xi^i_{1:t}\right)}{q_t\left(\xi^i_{1:t}\right)}.$$

The IS estimator $\pi^N_t\left(\phi_t\right)$ shares with the Monte Carlo estimator of the previous subsection the important properties of being an unbiased estimator of $\pi_t\left(\phi_t\right)$ and again the strong law of large numbers ensures that it converges almost surely to $\pi_t\left(\phi_t\right)$ as $N \longrightarrow \infty$. In many non trivial cases we are able to evaluate π_t only up to an unknown normalising constant Z_t, and when this is the case even this method turns out to be inapplicable.

A possible solution is the introduction of a set of self-normalised weights. The idea is to approximate separately the numerator and the denominator in the decomposition of the target distribution density $\pi_t\left(x_{1:t}\right) = \frac{\gamma_t(x_{1:t})}{Z_t}$. First we approximate the denominator Z_t exploiting the identity

$$Z_t = \int_{\mathsf{X}^t}\gamma_t\left(x_{1:t}\right)dx_{1:t} = \int_{\mathsf{X}^t}W_t\left(x_{1:t}\right)q_t\left(x_{1:t}\right)dx_{1:t} \tag{3.1}$$

where $W_t\left(x_{1:t}\right) = \frac{\gamma_t(x_{1:t})}{q_t(x_{1:t})}$ is the unnormalised weight function. Given N independent identically distributed samples $\xi^i_{1:t} \sim q_t$ the Monte Carlo approximation of Z_t is given by

$$Z^N_t = \frac{1}{N}\sum_{i=1}^{N}W_t\left(\xi^i_{1:t}\right). \tag{3.2}$$

It is worth noticing that Z^N_t is a unbiased and consistent estimate of Z_t, which is the main quantity of interests in many applications, including those discussed in this thesis. The unnormalised measure associated with the density γ_t can be approximated with the empirical measure

$$\gamma^N_t\left(dx_{1:t}\right) = \frac{1}{N}\sum_{i=1}^{N}W_t\left(\xi^i_{1:t}\right)\delta_{\xi^i_{1:t}}\left(dx_{1:t}\right).$$

The ratio of the two approximations is

$$\pi^N_t\left(dx_{1:t}\right) = \frac{\frac{1}{N}\sum_{i=1}^{N}W_t\left(\xi^i_{1:t}\right)\delta_{\xi^i_{1:t}}\left(dx_{1:t}\right)}{\frac{1}{N}\sum_{i=1}^{N}W_t\left(\xi^i_{1:t}\right).} = \sum_{i=1}^{N}w^i_t\delta_{\xi^i_{1:t}}\left(dx_{1:t}\right) \tag{3.3}$$

where $w_t^i = \frac{W_t\left(\xi_{1:t}^i\right)}{\sum_{k=1}^N W_t\left(\xi_{1:t}^k\right)}$ for $i \in 1:N$ are the self-normalised weights. Being the ratio of two unbiased estimator does not guarantee that π_t^N is unbiased, which indeed is not the case in general. However $\pi_t^N\left(\phi_t\right)$ is strongly consistent for any integrable test function ϕ_t, subject to $\pi\left(\phi_t^2\right) < \infty$, and it satisfies a Central Limit Theorem of which we will state a general form in section 3.5.

3.3 Sequential importance sampling

At the beginning of this chapter we introduced SMC methods as a tool to approximate a sequence of target probability densities $\left(\pi_t\right)_{t\in 1:T}$. In order to do so sequentially we need to select a proposal distribution with the following structure

$$q_t\left(x_{1:t}\right) = \mu_1\left(x_1\right)\prod_{s=2}^{t} f_s\left(x_{s-1}, x_s\right)$$

for all $t \in 1:T$. When it is possible to do so, we can obtain a particle $\xi_{1:t}^i \sim q_t$ first by sampling $\xi_1^i \sim \mu_1$ at time $t = 1$ and then propagating the particle by sampling $\xi_s^i \sim f_s\left(\xi_{s-1}^i, \cdot\right)$ for $s \in 2:t$. The unnormalised weight function $W_t\left(x_{1:t}\right)$ admits the recursive decomposition

$$\begin{aligned} W_t\left(x_{1:t}\right) &= \frac{\gamma_t\left(x_{1:t}\right)}{q_t\left(x_{1:t}\right)} = \frac{\gamma_{t-1}\left(x_{1:t-1}\right)}{q_{t-1}\left(x_{1:t-1}\right)}\frac{\gamma_t\left(x_{1:t}\right)}{\gamma_{t-1}\left(x_{1:t-1}\right) f_t\left(x_{t-1}, x_t\right)} \\ &= W_{t-1}\left(x_{t-1}\right)\alpha_t\left(x_{1:t}\right) \end{aligned}$$

where the incremental importance weight function α_t is given by

$$\alpha_t\left(x_{1:t}\right) = \frac{\gamma_t\left(x_{1:t}\right)}{\gamma_{t-1}\left(x_{1:t-1}\right) f_t\left(x_{t-1}, x_t\right)}$$

for all $t \in 1:T$. The resulting sequential importance sampling algorithm proceeds as follows.

At any time step $t \in 1:T$ estimates for the target distribution π_t and for the normalising constant Z_t are given by equations (3.3) and (3.2) respectively. The advantage of the SIS algorithm is the reduced cost of propagating the particle system $\xi_{1:t}^{1:N}$ with respect to sampling a set of completely new particles. When the computational cost of sampling from $f_t\left(x_{t-1}, \cdot\right)$ and evaluating the

Algorithm 3.1 Sequential Importance Sampling

```
1. Sample $\xi_1^i \sim q_1$ independently for $i \in 1 : N$. Set $w_t^i \propto W_1\left(\xi_1^i\right)$ for $i \in 1 : N$.
2. For $t = 2, \ldots, T$,
    (a) Sample independently $\xi_t^i \sim f_t\left(\xi_{t-1}^i, \cdot\right)$ for $i \in 1 : N$.
    (b) Set $w_t^i \propto W_t\left(\xi_{1:t}^i\right) = W_{t-1}\left(\xi_{1:t-1}^i\right) \alpha_t\left(\xi_{1:t}^i\right)$ for $i \in 1 : N$.
```

incremental weights $\alpha_t\left(x_{1:t}\right)$ is independent of t, which is often the case, the computational complexity of the SIS algorithm is fixed at any time step. On the contrary an algorithm that samples exactly from π_t sequentially at each time t would have a computational complexity at time t increasing at least linearly with t. When the incremental weight $\alpha_t\left(x_{1:t}\right)$ does not depend on $x_{1:t-1}$ we write $g_t\left(x_t\right) = \alpha_t\left(x_{1:t}\right)$. This is a desirable property for the incremental weights as in this case the domain of g_t does not increase in dimension with t, therefore in usual scenarios we can evaluate g_t at a fixed computational cost. From now on we will consider propagation mechanisms of this type and in Part 2 we develop an algorithm such that the incremental weights have this property.

As SIS is just a standard IS that exploits a particular decomposition of the proposal distribution, it inherits the same drawbacks, in particular a variance of the estimates that typically grows exponentially in time. A way to address this problem is the introduction of some form of interaction within particles, namely the resampling methods.

3.4 Sequential Monte Carlo

With SIS at any time step t we have a weighted set of particles $\left(\xi_{1:t}^{1:N}, w_t^{1:N}\right)$ approximating π_t. While the weighted particles provide approximations to integrals of test functions straightforwardly, to obtain an approximate sample from the distribution π_t we need to perform a further sampling step. This consists of drawing a sample from the IS approximation π_t^N of π_t which is itself obtained by sampling: for this reason this operation is called resampling. In order to obtain such sample of size N, we can draw independently N indices

$A_t^1, \dots, A_t^N$ from a Categorical distribution Cat $\left(w_t^{1:N}\right)$, that is the distribution with outcome space $\mathsf{S} = \{1, 2, \dots, N\}$ and such that

$$\mathbb{P}\left(A_t^i = k\right) = w_t^k$$

for all $i \in 1 : N$ and $k \in \mathsf{S}$. The approximate sample from π_t is given by $\left(\xi_{1:t}^{A_t^1}, \dots, \xi_{1:t}^{A_t^N}\right)$. In this thesis we adopt this resampling procedure in all the simulations involving SMC algorithms. An overview of alternative resampling schemes such as stratified resampling and systematic resampling can be found in Cappé et al. (2007). As resampling at each time step can be detrimental in terms of the quality of the SMC estimates, a common practice we adopt is the use of an effective sample size-based adaptive resampling scheme (Kong et al. 1994, Liu & Chen 1995), about which we provide more details in Section 7.1. The idea of SMC is to incorporate the resampling step within the propagation mechanism. At each time step instead of propagating according to the chosen proposal distribution q_t the set of weighted particles, we propagate an equal sized set of unweighted particles obtained by resampling.

Algorithm 3.2 A Particle Filter

```
1. Sample $\xi_1^i \sim \mu_1$ independently for $i \in \{1, \dots, N\}$.
2. For $t = 2, \dots, T$, sample independently
       $\xi_t^i \sim \frac{\sum_{j=1}^N g_{t-1}(\xi_{t-1}^j) f_t(\xi_{t-1}^j, \cdot)}{\sum_{j=1}^N g_{t-1}(\xi_{t-1}^j)}, \qquad i \in \{1, \dots, N\}.$
```

Running Algorithm 3.2 with

$$\mu_1 = \mu, \qquad f_t = f, \qquad g_t(x) = g(x, y_t), \tag{3.4}$$

corresponds exactly to running the bootstrap particle filter of Gordon et al. (1993), and we observe that when (3.4) holds, the quantity Z_T defined in (3.1) is identical to L, so that Z_T^N defined in (3.2) is an approximation of L.

3.5 A central limit theorem for the normalising constant estimator

A wide range of convergence results is available for SMC algorithms in Del Moral (2004). As our main statistical quantity of interest is $Z_T := \mathbb{E}\left[\prod_{t=1}^{T} g_t\left(X_t\right)\right]$, which in an HMM setting corresponds to the marginal likelihood L associated with $y_{1:T}$, we provide here a Central Limit Theorem for its estimator obtained from Algorithm (3.2). This is an adaptation to our notation and setting of the convergence result for SMC that can be found in Doucet & Johansen (2011). An elegant proof of this result is given in Del Moral (2004, Chapter 9). Note that given a particle system $\left\{\xi_{1:T}^{1:N}\right\}$ evolving according to Algorithm 3.2

$$Z_T^N := \prod_{t=1}^{T} \frac{1}{N} \sum_{i=1}^{N} g_t\left(\xi_t^i\right)$$

is an unbiased estimator of Z_T (see for example Doucet & Johansen (2011)).

Proposition 1. *For the estimator Z_T^N we have*

$$\sqrt{N}\left(\frac{Z_T^N}{Z_T} - 1\right) \xrightarrow{d} \mathcal{N}(0, \sigma^2),$$

where $\xrightarrow{d}$ denotes convergence in distribution and

$$\sigma^2 := \int_{\mathsf{X}} \frac{\pi_T^2\left(x_1\right)}{\mu_1\left(x_1\right)} dx_1 - 1 + \sum_{t=2}^{T}\left(\int_{\mathsf{X}^t} \frac{\pi_T^2\left(x_{1:t}\right)}{\pi_{t-1}\left(x_{1:t-1}\right) f_t\left(x_{t-1}, x_t\right)} dx_{1:t} - 1\right).$$

Note that we used the simplified notation $\pi_T\left(x_{1:t}\right) := \int_{\mathsf{X}^{T-t}} \pi_T\left(x_{1:T}\right) dx_{t+1:T}$. Given a sequence of target distributions $\pi_{1:T}$, a sensible approach when selecting an appropriate sequence of proposals $\mu_1, f_2, \ldots, f_T$ is trying to minimise the asymptotic variance σ^2. In Section 6.4 we will show how to retrieve a consistent estimator $\pi_T^N\left(\phi_T\right)$ for the quantity $\pi_T\left(\phi_T\right)$ from Algorithm 3.2, for a square integrable test function ϕ_T on X^T. Although an analogous asymptotic result is available for the estimator $\pi_T^N\left(\phi_T\right)$ (see Doucet & Johansen (2011)) and we provide it below with Proposition 2, trying to minimise the asymptotic variance of $\pi_T^N\left(\phi_T\right)$ when designing a particle filter could prove detrimental, as this depends on the specific test function ϕ_T whereas we are typically interested in the expectations of several test functions.

Proposition 2. *For the estimator $\pi_T^N(\phi_t)$ we have*

$$\sqrt{N}\left(\pi_T^N(\phi_T) - \pi_T(\phi_T)\right) \xrightarrow{d} \mathcal{N}\left(0, \sigma^2\right),$$

where $\xrightarrow{d}$ denotes convergence in distribution and

$$\begin{aligned}
\sigma^2 \quad := & \quad \int_{\mathsf{X}} \frac{\pi_T^2(x_1)}{\mu_1(x_1)} \left[\int_{\mathsf{X}} \phi_T(x_{1:T})\, \pi_T(x_{2:T} \mid x_1)\, dx_{2:T} - \pi_T(\phi_T)\right]^2 dx_1 \\
+ & \quad \sum_{t=2}^{T-1} \int_{\mathsf{X}^t} \frac{\pi_T^2(x_{1:t})}{\pi_{t-1}(x_{1:t-1})\, f_t(x_{t-1}, x_t)} \\
& \quad \cdot \left[\int_{\mathsf{X}} \phi_T(x_{1:T})\, \pi_T(x_{t+1:T} \mid x_{1:t})\, dx_{t+1:T} - \pi_T(\phi_T)\right]^2 dx_{1:t} \\
+ & \quad \int_{\mathsf{X}^\mathsf{T}} \frac{\pi_T^2(x_{1:T})}{\pi_{T-1}(x_{1:T-1})\, f_T(x_{T-1}, x_T)} \left[\phi_T(x_{1:T}) - \pi_T(\phi_T)\right]^2 dx_{1:T}.
\end{aligned}$$

Chapter 4

Background

SMC methods were originally designed for online filtering, which refers to the problem of approximating subsequent filtering distributions in real time as information becomes available. In this thesis we focus on an offline setting, where the sequence of observations $y_{1:T}$ is fixed, and on the related problems of estimating the normalising constant Z_T or some function expectations under the smoothing distribution. Some important applications present an offline nature such as those involving the particle marginal Metropolis–Hastings (PMMH) method of Andrieu et al. (2010) and the importance sampling squared (IS2) method of Tran et al. (2014). When implementing a particle filter for an offline setting, we can get important gains in terms of algorithm efficiency if we design the particle evolution dynamic at time t based on the full observation sequence $y_{1:T}$ rather than just on the information $y_{1:t}$ up to time t. In Section 4.1 we give a brief review of look-ahead methods, the class of SMC methods the iAPF belongs to, which are based on this strategy.

An alternative approach with respect to SMC methods in the offline setting is to use Markov chain Monte Carlo (MCMC) or reversible jump MCMC methods (Gamerman & Lopes (2006), Green (1995)). MCMC methods are extremely popular, however their efficient implementation can be challenging in complex problems, where a careful design of proposal densities is required. Instances where non-MCMC methods can be more efficient than MCMC algorithms include time-ordered hidden Markov models (Chopin (2007)) and mixture models (Del Moral et al. (2006), Fearnhead (2004)) (for a longer list of applications and discussion see Fearnhead (2008)).

4.1 Look-ahead methods

Given a sequence of target distribution densities $\pi_{1:T}$ where π_t is defined on the state space X^t, Proposition 1 of Chapter 3 and related results suggest that the choice of proposal distribution densities μ_1 and $f_{2:T}$ is critical in controlling the variance of the SMC estimator. In the typical HMM framework, the sequence of target distributions $\pi_{1:T}$ takes the form of a sequence of conditional densities $\pi_t\left(x_{1:t}\right) = p\left(x_{1:t} \mid y_{1:t}\right)$ for $t \in 1:T$ given the stream of information that consists of a sequence of observations $y_{1:T}$. In this setting even if the proposal distributions μ_1 and f_t are carefully designed for all $t \in 2:T$, standard SMC methods can perform poorly for example when two consecutive target distributions $p\left(x_{1:t} \mid y_{1:t}\right)$ and $p\left(x_{1:t+1} \mid y_{1:t+1}\right)$ present a high discrepancy. In this case the variance of the incremental weights at time $t+1$ is likely to be high, leading to the degeneracy of the overall SMC importance weights. When an HMM is used to model a dynamic system which posses strong memory, future information can help sharpen the inference about current state and mitigate the aforementioned problem. This can be the case for target tracking systems such as in Godsill & Vermaak (2004), Ikeda & Watanabe (1981), and protein structure prediction such as in Zhang & Liu (2002) for example. Look-ahead methods constitute a subset of SMC methods that follows the basic principle of using "future" information for designing the behaviour of the particle system $\left\{\xi_{1:T}^{1:N}\right\}$.

Consider the sequence of target distribution densities $\pi_t\left(x_{1:t}\right) = p\left(x_{1:t} \mid y_{1:T}\right)$ for $t \in 1:T$. In this case all the information up to the end of the sequence of observations $y_{1:T}$ is perfectly conveyed in the target distribution densities. If we can use the sequence of proposals $\mu_1\left(x_1\right) = p\left(x_1 \mid y_{1:T}\right)$ and $f_t\left(x_{t-1}, x_t\right) = p\left(x_t \mid x_{t-1}, y_{t:T}\right)$, for the corresponding incremental weight function $\alpha_t\left(x_{1:t}\right)$ we have

$$\begin{aligned}\alpha_t\left(x_{1:t}\right) &\propto \frac{\pi_t\left(x_{1:t}\right)}{\pi_{t-1}\left(x_{1:t-1}\right) f_t\left(x_{t-1}, x_t\right)} \\ &= \frac{p\left(x_{1:t} \mid y_{1:T}\right)}{p\left(x_{1:t-1} \mid y_{1:T}\right) p\left(x_t \mid x_{t-1}, y_{t:T}\right)} \\ &= 1\end{aligned}$$

for all $t \in 2:T$ and similarly for the initial proposal density

$$\alpha_1(x_1) \propto \frac{\pi_1(x_1)}{\mu_1(x_1)} = \frac{p(x_1 \mid y_{1:T})}{p(x_1 \mid y_{1:T})} = 1.$$

For this particle dynamic the variance of the particle weights is minimised (i.e. equal to zero) and resampling is unnecessary. Equivalently the asymptotic variance of the normalising constant estimator Z_T^N is minimised being itself equal to zero as Proposition 1 of Chapter 3 shows. This is an ideal scenario that we will refer to as *optimal look-ahead particle filter* and it gives some simple motivational basis for look-ahead strategies. We call the aforementioned proposals densities μ_1 and f_t for $t \in 1:T$ optimal look-ahead proposals. In most realistic scenarios it is not possible to implement the optimal look-ahead particle filter. If the state space X is not finite we do not have a general method to sample particles according to the optimal look-ahead proposals $p(x_1 \mid y_{1:T})$ and $p(x_t \mid x_{t-1}, y_{t:T})$ for $t \in 2:T$. This particle dynamic can be implemented only in a restricted number of models such as the linear Gaussian model (thanks to the Kalman filter and related techniques). Also the integration operation which is typically necessary to evaluate $p(x_{1:t} \mid y_{1:T}) = \int_{\mathsf{X}^{T-t}} p(x_{1:T} \mid y_{1:T})\, dx_{t:T}$ for different $t \in 1:T$ as it appears in the incremental weights functions is in general unfeasible. If the state space X is finite it is possible to perform these operations but at a computational cost which is typically unfeasible for the cases of interests where X is relatively large.

Another important idealised scenario is when the sequence of target distribution densities takes the form of

$$\pi_t(x_{1:t}) = p(x_{1:t} \mid y_{1:t+1})$$

and the proposal distributions are

$$\mu_1(x_1) = p(x_1 \mid y_1) \text{ and } f_t(x_{t-1}, x_t) = p(x_t \mid x_{t-1}, y_t).$$

For the correspondent incremental weights we have that

$$\begin{aligned} \alpha_t(x_{1:t}) &= \frac{\pi_t(x_{1:t})}{\pi_{t-1}(x_{1:t-1}) f_t(x_{t-1}, x_t)} \\ &= \frac{p(x_{1:t} \mid y_{1:t+1})}{p(x_{1:t-1} \mid y_{1:t}) p(x_t \mid x_{t-1}, y_t)} \\ &= \frac{p(x_{1:t-1} \mid y_{1:t}) \frac{p(x_t \mid x_{t-1}, y_t) p(y_{t+1} \mid x_t)}{p(y_{t+1} \mid y_t)}}{p(x_{1:t-1} \mid y_{1:t}) p(x_t \mid x_{t-1}, y_t)} \\ &\propto p(y_{t+1} \mid x_t). \end{aligned}$$

This is known as the *fully adapted particle filter.* The fully adapted particle filter is described in a slightly different framework in Pitt & Shephard (1999) where it is seen as the particle filter given from the standard sequence of target distribution densities $\pi_t(x_{1:t}) = p(x_{1:t} \mid y_{1:t})$ and the sequence of proposals $\mu_1(x_1) = p(x_1 \mid y_1)$ and $f_t(x_{t-1}, x_t) = p(x_t \mid x_{t-1}, y_t)$. The corresponding incremental weights take the form of $\alpha_t(x_{1:t}) = p(y_t \mid x_{t-1})$ and therefore they are independent from the outcome of the sampling step at time t. In this case we can perform the resampling step before rather than after the sampling step, improving the overall efficiency of the particle filter. The description of the fully adapted particle filter and of other particle filters under the framework of algorithm 3.2 is due to Johansen & Doucet (2008). In general for many models of interest it is either impossible or computationally unfeasible to implement the fully adapted particle filter for the same reasons that we mentioned regarding the optimal look-ahead particle filter. A possible approach to particle filtering inspired by the fully adapted particle filter and which has been very successful is originally due to Pitt & Shephard (1999) and is known under the name of *auxiliary particle filter* (APF). Here we briefly provide the improved version of the original APF due to Carpenter et al. (1999) which only includes one resampling step at each time instance, as presented in Johansen & Doucet (2008).

The APF described in Carpenter et al. (1999) corresponds to the particle filter given from the sequence of target densities

$$\pi_t(x_{1:t}) = \hat{p}(x_{1:t} \mid y_{1:t+1}) \propto p(x_{1:t} \mid y_{1:t}) \hat{p}(y_{t+1} \mid x_t),$$

where $\hat{p}(y_{t+1} \mid x_t)$ is a suitable approximation of $p(y_{t+1} \mid x_t)$, and from the sequence of proposal densities

$$\mu_1\left(x_1\right)=\hat{p}\left(x_1 \mid y_1\right) \text{ and } f_t\left(x_{t-1}.x_t\right)=\hat{p}\left(x_t \mid x_{t-1},y_t\right)$$

where $\hat{p}\left(x_1 \mid y_1\right)$ and $\hat{p}\left(x_t \mid x_{t-1},y_t\right)$ are typically approximations of $p\left(x_1 \mid y_1\right)$ and $p\left(x_t \mid x_{t-1},y_t\right)$ respectively for $t \geq 2$. When the approximations of such target and proposal densities are exact, we obtain the perfect adaption, that is the fully adapted particle filter. Note that in this case at time t we do not approximate directly the distribution density $p\left(x_{1:t} \mid y_{1:t}\right)$ or a smoothing distribution density $p\left(x_{1:t} \mid y_{1:T}\right)$. In order to do so we can use importance sampling with the importance distribution

$$\hat{p}\left(x_{1:t-1} \mid y_{1:t}\right)\hat{p}\left(x_t \mid x_{t-1},y_t\right)=p\left(x_{1:t-1} \mid y_{1:t-1}\right)\hat{p}\left(y_t \mid x_{t-1}\right)\hat{p}\left(x_t \mid x_{t-1},y_t\right)$$

whose approximation is obtained from the set of equally weighted particles after the sampling step at time t and before the resampling step. For the implementation of the APF the critical point is the design of the approximations $\hat{p}\left(x_t \mid x_{t-1},y_t\right)$ and $\hat{p}\left(y_t \mid x_t\right)$. In general we should select a $\hat{p}\left(x_t \mid x_{t-1},y_t\right)$ with thicker tails than $p\left(x_t \mid x_{t-1},y_t\right)$ in order to have a bounded incremental weights function α_t for all $t \in 1:T$. Furthermore if we use $\hat{p}\left(x_{1:t-1} \mid y_{1:t}\right)\hat{p}\left(x_t \mid x_{t-1},y_t\right)$ as an importance sampling distribution to retrieve an approximation to $p\left(x_{1:t} \mid y_{1:t}\right)$ for the relative importance weights we have that

$$\begin{aligned}\frac{p\left(x_{1:t} \mid y_{1:t}\right)}{\hat{p}\left(x_{1:t-1} \mid y_{1:t}\right)\hat{p}\left(x_t \mid x_{t-1},y_t\right)} &\propto \frac{p\left(x_{1:t-1} \mid y_{1:t-1}\right)p\left(y_t \mid x_{t-1}\right)p\left(x_t \mid x_{t-1},y_t\right)}{p\left(x_{1:t-1} \mid y_{1:t-1}\right)\hat{p}\left(y_t \mid x_{t-1}\right)\hat{p}\left(x_t \mid x_{t-1},y_t\right)}\\ &= \frac{p\left(y_t \mid x_{t-1}\right)}{\hat{p}\left(y_t \mid x_{t-1}\right)}\cdot\frac{p\left(x_t \mid x_{t-1},y_t\right)}{\hat{p}\left(x_t \mid x_{t-1},y_t\right)}\end{aligned}$$

therefore we also want the function $\hat{h}\left(x_t\right)=\hat{p}\left(y_{t+1} \mid x_t\right)$ to have thicker tails than $h\left(x_t\right)=p\left(y_{t+1} \mid x_t\right)$ for these importance weights to be upper bounded. A possible approach (Pitt & Shephard (1999)) for the design of $\hat{p}\left(y_t \mid x_{t-1}\right)$ is to set $\hat{p}\left(y_{t+1} \mid x_t\right)=p\left(y_{t+1} \mid m\left(x_t\right)\right)$ where $m\left(x_t\right)$ corresponds to the mean, mode or median of $f\left(x_t,\cdot\right)$. Although this simple approach is often applicable given tractable model transitions, it does not guarantee for the incremental functions and for the importance weights to have the aforementioned property of being upper bounded.

Given an HMM (μ, f, g) the optimal look-ahead particle filter and the fully adapted particle filter have the property that they do not extend the domain of the functions μ, f and g in the sense that μ_1 and f_t are defined on the same state space X as μ and f for all $t \in 2 : T$ and the potential functions (i.e. incremental weights functions) g_t are defined on the same state space Y as g for all $t \in 1 : T$. This feature is also shared by the $\psi - APF$, the particle filter that we will introduce in Section 5.1, and it is essential in order for these particle filters to be described under the framework of algorithm 3.2. This property is not shared by, other than the APF, the *block sampling* and *pilot look-ahead sampling*, which are two look-ahead strategies well known in the literature (see for a more detailed exposition of these look-ahead algorithms Lin et al. (2013)).

Block sampling is a look-ahead strategy proposed in Doucet et al. (2006) which consists of enlarging the state space of the proposals μ_1 and f_t in order to sample and update sections of the particles paths over a fixed lag L. With this method also the dimensionality of the target distribution π_t is enlarged, in a way such that by construction π_t admits the required distribution $p\left(x_{1:t} \mid y_{1:t}\right)$ as a marginal. These extended distributions are designed to circumvent the problem of computing integrals which do not admit a closed-form expression such as those involved in the implementation of the optimal look-ahead particle filter. The optimal choice for the block sampling proposals μ_1 and f_t are $\mu_1\left(x_{1:1+L} \mid y_{1:1+L}\right)$ and $f_t\left(x_{t-1}, x_{t:t+L}\right) = p\left(x_{t:t+L} \mid x_{t-1}, y_{t:t+L}\right)$ for $t \geq 2$ but as typically these sampling distributions are inaccessible, in practice we can use approximations $\hat{p}\left(x_{1:1+L} \mid y_{1:1+L}\right)$ and $\hat{p}\left(x_{t:t+L} \mid x_{t-1}, y_{t:t+L}\right)$ of the optimal choices $p\left(x_{1:1+L} \mid y_{1:1+L}\right)$ and $p\left(x_{t:t+L} \mid x_{t-1}, y_{t:t+L}\right)$ respectively. The block sampling approach can be viewed as a natural extension of the APF corresponding to the case $L = 0$, and it outperforms standard SMC methods in the applications presented in Doucet et al. (2006). However the performance of this approach depends entirely on the ability of the user to design good approximations $\hat{p}\left(x_{1:1+L} \mid y_{1:1+L}\right)$ and $\hat{p}\left(x_{t:t+L} \mid x_{t-1}, y_{t:t+L}\right)$ of the optimal distributions. Although some techniques for the design of efficient proposal distributions can be applied to this framework, there is no general method to construct an approximation $\hat{p}$ leading to an efficient block sampling look-ahead algorithm.

A pilot exploration method is proposed in Zhang & Liu (2002) in which at every time step t the space of future states up to time $t + L$ is partially explored by pilot particle paths in order to convey future information in the particles evolution dynamic. The method was introduced for the case of finite state space of $\mathsf{X} = \{a_1, \dots, a_M\}$. In this setting at every time step t and for every particle ξ_{t-1}^i, $i \in 1 : N$, M pilot paths each corresponding to a state a_j with $j \in 1 : M$ are generated independently according to a predefined pilot proposal distribution. A suitable auxiliary weight is assigned to each pilot path and each particle ξ_{t-1}^i is then propagated via a resampling step that takes into account such auxiliary weights. An alternative deterministic approach to the generation of the pilot paths is also studied in Zhang & Liu (2002). If no prior structural information is available the pilot proposal can correspond to the model transition distribution f, therefore with this approach the design of the proposal distribution is less critical with respect to block sampling approach. However this method presents some limitations. In particular when the state space X is continuous, it is not possible to explore all the possible values of x_t. One possible approach is to generate a sample of the state space from the model transition and treat it as the set of values that ξ_t can assume, then proceed as in the finite case. However producing a sample for each particle that effectively explores a large state space X is unfeasible in most scenarios. A more detailed but self contained description of pilot look-ahead sampling methods including deterministic piloting and multilevel piloting is contained in Lin et al. (2013).

An extremely recent look-ahead scheme contemporary to the present work is presented in Scharth & Kohn (2016) under the name of particle efficient importance sampling (P-EIS). The P-EIS algorithm is based on the efficient importance sampling algorithm (EIS) of Richard & Zhang (2007), which is an importance sampling method for the estimation of high-dimensional integrals that have a sequential structure. The P-EIS algorithm constructs a global approximation of a target proposal distribution by iterating a sequence of least-squares regressions, and this feature is shared by the iAPF. However the P-EIS algorithm and the iAPF are different in their essence. With the P-EIS algorithm least-squares regressions are used to define proposals and potentials of a particle filter which is then run once. With the iAPF, backward minimisation routines (that can be least-squares regressions) and particle filter iterations alternate dynamically and adaptively. Furthermore the iAPF uses a different form of least-squares regression that exploits the decomposition of the SMC proposal in the product of model transition and look-ahead functions.

4.2 Particle Marginal Metropolis–Hastings

Particle marginal Metropolis Hasting (PMMH) is a powerful parameter estimation method presented in Andrieu et al. (2010) that combines standard MCMC and SMC schemes. In important settings it makes it possible to build efficient high dimensional Metropolis–Hastings (MH) proposal distributions by using SMC methods and also to make Bayesian inference feasible for a large class of statistical models where this was not previously so. This method builds on a pseudo-marginal method originally introduced in Beaumont (2003) and further studied in Andrieu & Roberts (2009) where theoretical results describing the convergence properties of the original method and a modification of it are given. In particular in Andrieu & Roberts (2009) it is shown how pseudo-marginal algorithms share the same marginal stationary distribution as the idealised MH algorithm they are approximations of. For many applications in a HMM setting, for example those discussed in Part III, such approximations are based on estimates Z_T^N of the normalising constant Z_T. The accuracy of unbiased estimators of these quantities is both of critical importance for the overall performance of the a PMMH scheme (Andrieu & Vihola (2015), Lee & Łatuszyński (2014), Sherlock et al. (2015), Doucet et al. (2015)) and (partly because of this) the central motivation of the algorithm developed in Part II. We provide here a brief description of the PMMH method pertinent to our applications in part III.

Consider a HMM $(\mu_\theta, f_\theta, g_\theta)$ where the dynamic of the system depends on a parameter of interest $\theta \in \Theta \subseteq \mathbb{R}^n$. In a Bayesian framework, given a sequence of observations $y_{1:T}$ we focus on the parameter estimation problem of θ based on the posterior distribution density

$$\begin{aligned}\pi(\theta) &\propto \pi_{\text{prior}}(\theta)\, p_\theta(y_{1:T}) \\ &= \pi_{\text{prior}}(\theta)\, \mathbb{E}_\theta\left(\prod_{i=1}^{T} g_\theta(X_t, y)\right) \\ &= \pi_{\text{prior}}(\theta)\, Z_\theta\end{aligned}$$

defined on Θ and for a suitable prior distribution density π_{prior} on the parameter θ. Two other problems directly related to this are simulation according to $\pi(\theta)$ and computing expectations with respect to it. The Markov chain Monte Carlo method is a relatively general approach for dealing with such problems that consists of simulating an ergodic Markov chain $(\theta_i)_{i\geq 1}$ which admits $\pi(\theta)$ as invariant probability density (see Robert & Casella (2005)). When it is not possible to sample according to $\pi(\theta)$ but this density is known analytically or cheap to compute up to a normalising constant, a popular approach to the construction of such Markov chain $(\theta_i)_{i\geq 1}$ is the Metropolis–Hastings algorithm. This scheme dates back to Metropolis et al. (1953) and its adaptation to non symmetric kernels is due to Hastings (1970). The Metropolis-Hastings algorithm consists of the following procedure. Given the current state θ_i of the Markov chain at step i, we propose a value $\theta' \sim k(\theta_i, \cdot)$ sampled from an instrumental kernel $k : \Theta \times \Theta \longrightarrow \mathbb{R}^+$, which is then accepted with a certain probability $\alpha(\theta_i, \theta')$ which depends on the values of the densities $\pi(\theta_i)$ and $\pi(\theta')$. If the value is accepted then we set $\theta_{i+1} = \theta'$, otherwise we set $\theta_{i+1} = \theta_i$. The procedure is described with pseudo-code in Algorithm 4.1, for a given length L of the Metropolis–Hastings chain.

Algorithm 4.1 Metropolis–Hastings algorithm

```
1. Set $\theta_1 = \theta_0 \in \Theta$
2. For $i \in 1 : L$
    (a) Sample $\theta' \sim k(\theta_i, \cdot)$
    (b) Compute
            $\alpha(\theta_i, \theta') = 1 \wedge \frac{\pi(\theta') k(\theta', \theta_i)}{\pi(\theta_i) k(\theta_i, \theta')}$
    (c) With probability $\alpha(\theta_i, \theta')$ set $\theta_{i+1} = \theta'$. Otherwise set $\theta_{i+1} = \theta_i$.
```

If the kernel $k(\theta, \theta')$ is strictly positive for every $\theta, \theta' \in \text{supp}(\pi)$ the resulting Markov chain $(\theta_i)_{i\geq 1}$ is irreducible and has the desired invariant distribution π. Milder conditions on the irreducibility of the resulting Markov chain are given in Roberts & Tweedie (1996).

When $\pi(\theta)$ is analytically intractable or too complex to evaluate even up to a normalising constant, Algorithm 4.1 cannot be implemented. However we can consider approximations of this idealised algorithm. This idea is exploited in Beaumont (2003), Andrieu & Roberts (2009) where the density values $\pi(\theta_i)$ and $\pi(\theta')$ that appear in the acceptance probability of Algorithm 4.1 are approximated with importance sampling. Specifically for the HMM setting, when $p_\theta(y_{1:T})$ can be conveniently described as the marginal density of the extended distribution density $p_\theta(x_{1:T}, y_{1:T})$, we can derive an estimate Z_θ^N of $p_\theta(y_{1:T}) = Z_\theta$ using the output of an SMC algorithm with N particles targeting the distribution density $\pi_T(x_{1:T}) = p_\theta(x_{1:T} \mid y_{1:T})$. The resulting PMMH algorithm is proposed in Andrieu et al. (2010) and we present a simple pseudo-code description in Algorithm 4.2.

Algorithm 4.2 Particle Marginal Metropolis–Hastings algorithm

```
1. Initialisation
    (a) Set $\theta_1 = \theta_0 \in \Theta$
    (b) Run a particle filter to get an estimate $Z_{\theta_1}^N$ of $p_{\theta_1}(y_{1:T})$
2. For $i > 1$, while a stopping rule is not met
    (a) Sample $\theta' \sim k(\theta_i, \cdot)$
    (b) Run a particle filter to get an estimate $Z_{\theta'}^N$ of $p_\theta(y_{1:T})$
    (c) Compute
            $$\alpha(\theta_i, \theta') = 1 \wedge \frac{\pi_{\text{prior}}(\theta') Z_{\theta'}^N k(\theta', \theta_i)}{\pi_{\text{prior}}(\theta_i) Z_{\theta_i}^N k(\theta_i, \theta')}$$
    (d) With probability $\alpha(\theta_i, \theta')$ set $\theta_{i+1} = \theta'$, $Z_{\theta_{i+1}}^N = Z_{\theta'}^N$. Otherwise set
        $\theta_{i+1} = \theta_i$, $Z_{\theta_{i+1}}^N = Z_{\theta_i}^N$.
```

The key feature of Algorithm 4.2 is that the transition kernel corresponding to 2.(a)-(d) leaves the target density of interest $\pi(\theta)$ invariant. Furthermore in Andrieu et al. (2010, Proposition 4.4) mild assumptions that guarantee convergence for any fixed $N \geq 1$ are formulated. These are the standard conditions on the support of the SMC proposals and of convergence of the corresponding idealised MH algorithm. Note that these conditions guarantee also the convergence of the appropriately defined associated extended chain $(\theta_i, x_{1:T}^i)_{i \geq 1}$ to the extended distribution density

$$\pi\left(\theta, x_{1:T}\right) = \pi_{\text{prior}}\left(\theta\right) p_{\theta}\left(x_{1:T} \mid y_{1:T}\right)$$

which is a stronger result compare to the convergence of the sole marginal $\pi\left(\theta\right)$.

Part II

The Iterated Auxiliary Particle Filter

Chapter 5

Twisted Models

In the following chapters we give motivational ground to the iterated auxiliary particle filter and provide a general iAPF algorithm and two possible implementations with full specifications. We do so following the exposition of our research work in Guarniero et al. (2016).

5.1 The $\boldsymbol{\psi}$-auxiliary particle filter

Given an HMM (μ, f, g) and a sequence of observations $y_{1:T}$, we introduce a family of alternative twisted models based on a sequence of real-valued, bounded, continuous and positive functions $\boldsymbol{\psi} := (\psi_1, \psi_2, \ldots, \psi_T)$. Letting, for a transition density f as in Section 2.1 and bounded continuous positive function ψ, $f(x, \psi) := \int_{\mathsf{X}} f\left(x, x'\right) \psi\left(x'\right) dx'$, we define a sequence of normalizing functions $(\tilde{\psi}_1, \tilde{\psi}_2, \ldots, \tilde{\psi}_T)$ on X by

$$\tilde{\psi}_t(x_t) := f\left(x_t, \psi_{t+1}\right)$$

for $t \in \{1, \ldots, T-1\}$, $\tilde{\psi}_T \equiv 1$, and a normalizing constant $\tilde{\psi}_0 := \int_{\mathsf{X}} \mu\left(x_1\right) \psi_1\left(x_1\right) dx_1$. We then define the twisted model via the following sequence of twisted initial and transition densities

$$\mu_1^{\boldsymbol{\psi}}(x_1) := \frac{\mu(x_1)\psi_1(x_1)}{\tilde{\psi}_0}, \qquad f_t^{\boldsymbol{\psi}}(x_{t-1}, x_t) := \frac{f\left(x_{t-1}, x_t\right)\psi_t\left(x_t\right)}{\tilde{\psi}_{t-1}\left(x_{t-1}\right)}, \quad t \in \{2, \ldots, T\}, \tag{5.1}$$

and the sequence of positive functions

$$g_1^{\boldsymbol{\psi}}\left(x_1\right) := g\left(x_1, y_1\right) \frac{\tilde{\psi}_1\left(x_1\right)}{\psi_1\left(x_1\right)} \tilde{\psi}_0, \qquad g_t^{\boldsymbol{\psi}}\left(x_t\right) := g\left(x_t, y_t\right) \frac{\tilde{\psi}_t\left(x_t\right)}{\psi_t\left(x_t\right)}, \quad t \in \{2, \ldots T\}, \tag{5.2}$$

which play the role of observation densities in the twisted model. Our interest in this family of twisted models is motivated by the following invariance result.

Proposition 3. *If $\boldsymbol{\psi}$ is a sequence of bounded, continuous and positive functions, and*

$$Z_{\boldsymbol{\psi}} := \int_{\mathsf{X}^T} \mu_1^{\boldsymbol{\psi}}\left(x_1\right) g_1^{\boldsymbol{\psi}}\left(x_1\right) \prod_{t=2}^{T} f_t^{\boldsymbol{\psi}}\left(x_{t-1}, x_t\right) g_t^{\boldsymbol{\psi}}\left(x_t\right) dx_{1:T},$$

then $Z_{\boldsymbol{\psi}} = L$.

Proof. We observe that

$$\begin{aligned} & \mu_1^{\boldsymbol{\psi}}\left(x_1\right) g_1^{\boldsymbol{\psi}}\left(x_1\right) \prod_{t=2}^{T} f_t^{\boldsymbol{\psi}}\left(x_{t-1}, x_t\right) g_t^{\boldsymbol{\psi}}\left(x_t\right) \\ = & \frac{\mu(x_1)\psi_1(x_1)}{\tilde{\psi}_0} g\left(x_1, y_1\right) \frac{\tilde{\psi}_1\left(x_1\right)}{\psi_1\left(x_1\right)} \tilde{\psi}_0 \cdot \prod_{t=2}^{T} \frac{f\left(x_{t-1}, x_t\right) \psi_t\left(x_t\right)}{\tilde{\psi}_{t-1}\left(x_{t-1}\right)} g\left(x_t, y_t\right) \frac{\tilde{\psi}_t\left(x_t\right)}{\psi_t\left(x_t\right)} \\ = & \mu\left(x_1\right) g_1\left(x_1\right) \prod_{t=2}^{T} f\left(x_{t-1}, x_t\right) g\left(x_t, y_t\right), \end{aligned}$$

where we use that $\tilde{\psi}_T \equiv 1$ in the last term of the telescoping product. The result follows. □

Note that, in connection with Section 3.3, the initial and transition densities defined by the equations 5.1 effectively define an importance proposal

$$q\left(x_{1:T}\right) = \mu_1^{\boldsymbol{\psi}}\left(x_1\right) \prod_{t=2}^{T} f_t^{\boldsymbol{\psi}}\left(x_{t-1}, x_t\right)$$

for the path $x_{1:T}$. In this sense $\prod_{t=1}^{T} g_t^{\boldsymbol{\psi}}\left(x_t\right)$, the product of the elements in the set of potentials defined by equations 5.2, provides an alternative likelihood so that its product with the proposal $q\left(x_{1:T}\right)$ is unchanged with respect to the original model, which is exactly what Proposition 3 shows.

From a methodological perspective, Proposition 3 makes clear a particular sense in which our main statistical quantity of interest

$$L := \int_{X^T} \mu\left(x_1\right) g\left(x_1, y_1\right) \prod_{t=2}^{T} f\left(x_{t-1}, x_t\right) g\left(x_t, y_t\right) dx_{1:T},$$

the marginal likelihood associated with $y_{1:T}$, is common to an entire family of μ_1, $(f_t)_{t\in\{2,\ldots,T\}}$ and $(g_t)_{t\in\{1,\ldots,T\}}$. The BPF associated with the twisted model corresponds to choosing $\psi_t = 1$ for $i \in 1 : T$ so that

$$\mu^{\boldsymbol{\psi}} = \mu_1, \qquad f_t^{\boldsymbol{\psi}} = f_t, \qquad g_t^{\boldsymbol{\psi}} = g_t, \tag{5.3}$$

in Algorithm 3.2. To emphasize the dependence on $\boldsymbol{\psi}$, we provide in Algorithm 5.1 the corresponding algorithm and we will denote approximations of L by $Z_{\boldsymbol{\psi}}^N$. We demonstrate below that the BPF associated with the twisted model can also be viewed as an APF associated with the sequence $\boldsymbol{\psi}$, and so refer to this algorithm as the $\boldsymbol{\psi}$-APF. Since the class of $\boldsymbol{\psi}$-APFs is very large, it is natural to consider whether there is an optimal choice of $\boldsymbol{\psi}$, in terms of the accuracy of the approximation $Z_{\boldsymbol{\psi}}^N$: the following Proposition describes such a sequence.

Algorithm 5.1 $\boldsymbol{\psi}$-Auxiliary Particle Filter

```
1. Sample $\xi_1^i \sim \mu^{\boldsymbol{\psi}}$ independently for $i \in \{1, \ldots, N\}$.
2. For $t = 2, \ldots, T$, sample independently
       $\xi_t^i \sim \frac{\sum_{j=1}^N g_{t-1}^{\boldsymbol{\psi}}(\xi_{t-1}^j) f_t^{\boldsymbol{\psi}}(\xi_{t-1}^j, \cdot)}{\sum_{j=1}^N g_{t-1}^{\boldsymbol{\psi}}(\xi_{t-1}^j)}, \qquad i \in \{1, \ldots, N\}.$
```

Proposition 4. *Let* $\boldsymbol{\psi}^* := (\psi_1^*, \ldots, \psi_T^*)$*, where* $\psi_T^*\left(x_T\right) := g(x_T, y_T)$*, and*

$$\psi_t^*\left(x_t\right) := g\left(x_t, y_t\right) \mathbb{E}\left[\prod_{p=t+1}^{T} g\left(X_p, y_p\right) \middle| \{X_t = x_t\}\right], \quad x_t \in \mathsf{X}, \tag{5.4}$$

for $t \in \{1, \ldots, T-1\}$*. Then,* $Z_{\boldsymbol{\psi}^*}^N = L$ *with probability* 1.

Proof. It can be established that

$$g(x_t, y_t)\tilde{\psi}_t^*(x_t) = \psi_t^*(x_t), \qquad t \in \{1, \dots, T\}, \qquad x_t \in \mathsf{X}, \tag{5.5}$$

as this is trivially true for $t = T$ and easily shown for $t < T$ proceeding backwards. Therefore we obtain from (5.2) that $g_1^{\boldsymbol{\psi}^*} \equiv \tilde{\psi}_0^*$ and $g_t^{\boldsymbol{\psi}^*} \equiv 1$ for $t \in \{2, \dots, T\}$. Hence,

$$Z_N^{\boldsymbol{\psi}^*} = \prod_{t=1}^{T} \left[\frac{1}{N} \sum_{i=1}^{N} g_t^{\boldsymbol{\psi}^*} \left(\xi_t^i\right) \right] = \tilde{\psi}_0^*,$$

with probability 1. To conclude, we observe that

$$\begin{aligned} \tilde{\psi}_0^* &= \int_{\mathsf{X}} \mu\left(x_1\right) \psi_1^*\left(x_1\right) dx_1 = \int_{\mathsf{X}} \mu\left(x_1\right) \mathbb{E}\left[\prod_{t=1}^{T} g\left(X_t, y_t\right) \middle| \{X_1 = x_1\} \right] dx_1 \\ &= \mathbb{E}\left[\prod_{t=1}^{T} g\left(X_t, y_t\right) \right] = L. \quad \square \end{aligned}$$

In terms of the underlying Bayesian model, the optimal look-ahead functions and the corresponding optimal $\boldsymbol{\psi}^*$-APF have a straightforward interpretation. With the Bayesian notation the function ψ_t^* in equation 5.4 corresponds to

$$\psi_t^*\left(x_t\right) = p\left(y_{t:T} \mid x_t\right)$$

for all $t \in 1 : T$. Consequently for the normalising functions $\tilde{\psi}_t$ we have

$$\begin{aligned} \tilde{\psi}_t\left(x_t\right) &= \int_{\mathsf{X}} f\left(x_t, x_{t+1}\right) p\left(y_{t+1:T} \mid x_{t+1}\right) dx_{t+1} \\ &= p\left(y_{t+1:T} \mid x_t\right) \end{aligned}$$

for $t \in 1 : T-1$, with $\psi_T \equiv 1$ and $\tilde{\psi}_0 = \int_{\mathsf{X}} \mu\left(x_1\right) p\left(y_{1:T} \mid x_1\right) dx_1 = p\left(y_{1:T}\right)$. For the optimal twisted transitions/proposals we have that

$$\mu_1^{\boldsymbol{\psi}}\left(x_1\right) \propto \mu\left(x_1\right) p\left(y_{1:T} \mid x_1\right) \propto p\left(x_1 \mid y_{1:T}\right)$$

and

$$f_t^{\boldsymbol{\psi}}\left(x_{t-1}, x_t\right) \propto f\left(x_{t-1}, x_t\right) p\left(y_{t:T} \mid x_t\right) \propto p\left(x_t \mid x_{t-1}, y_{t:T}\right)$$

for $t \in 2 : T$, which are the transition densities that convey perfectly all the information up to the end of the observation sequence in the particles evolution dynamic. Note also that in these terms we have that

$$g\left(x_t, y_t\right) \tilde{\psi}_t\left(x_t\right) = p\left(y_t \mid x_t\right) p\left(y_{t+1:T} \mid x_t\right),$$

therefore equation 5.5 assumes the more intuitive form of

$$p\left(y_t \mid x_t\right) p\left(y_{t+1:T} \mid x_t\right) = p\left(y_{t:T} \mid x_t\right)$$

and it is easily verified given the structure of the Bayesian model.

Implementation of Algorithm 5.1 requires that one can sample according to $\mu_1^{\boldsymbol{\psi}}$ and $f_t^{\boldsymbol{\psi}}(x, \cdot)$ and compute $g_t^{\boldsymbol{\psi}}$ pointwise. This imposes restrictions on the choice of $\boldsymbol{\psi}$ in practice, since one must be able to compute both ψ_t and $\tilde{\psi}_t$ pointwise. In general models, the sequence $\boldsymbol{\psi}^*$ cannot be used for this reason as (5.4) cannot be computed explicitly. However, since Algorithm 5.1 is valid for any sequence of positive functions $\boldsymbol{\psi}$, we can interpret Proposition 4 as motivating the effective design of a particle filter by solving a sequence of function approximation problems. Furthermore other look-ahead methods rely on an implicit approximation of the optimal look-ahead function ψ^*. It has also been noted in Ruiz & Kappen (2017) that the computation of the optimal twisted functions present some similarities with the control literature, and in particular with the backward message passing involved in the computation of the optimal control solution in Kappen et al. (2012).

Alternatives to the BPF have been considered before (see, e.g., the "locally optimal" proposal in Doucet et al. 2000, the discussion in Del Moral 2004, Section 2.4.2 and the Background section in Part I of this thesis). The family of particle filters we have defined using $\boldsymbol{\psi}$ are unusual, however, in that $g_t^{\boldsymbol{\psi}}$ is a function only of x_t rather than (x_{t-1}, x_t). Other approaches in which the particles are sampled according to a transition density that is not f typically require the extension of the domain of these functions. This is again a consequence of the fact that the $\boldsymbol{\psi}$-APF can be viewed as a BPF for a twisted model. This feature is shared by the fully adapted APF of Pitt & Shephard (1999), when recast as a standard particle filter for an alternative model as in Johansen & Doucet (2008), and which is obtained as a special case of Algorithm 5.1 when

$\psi_t(\cdot) \equiv g(\cdot, y_t)$ for each $t \in \{1, \ldots, T\}$. We view the approach here as generalizing that algorithm for this reason.

It is possible to recover other existing methodological approaches as BPFs for twisted models. We have already seen that when each element of $\boldsymbol{\psi}$ is a constant function, we recover the standard BPF of Gordon et al. (1993). Setting $\psi_t(x_t) = g(x_t, y_t)$ gives rise to the fully adapted APF. By taking, for some $k \in \mathbb{N}$ and each $t \in \{1, \ldots, T\}$,

$$\psi_t(x_t) = g(x_t, y_t)\,\mathbb{E}\left[\prod_{p=t+1}^{(t+k)\wedge T} g(X_p, y_p) \,\middle|\, \{X_t = x_t\}\right], \quad x_t \in \mathsf{X}, \tag{5.6}$$

$\boldsymbol{\psi}$ corresponds to a sequence of look-ahead functions (see, e.g., Lin et al. 2013) and one can recover idealized versions of the delayed sample method of Chen et al. (2000) (see also the fixed-lag smoothing approach in Clapp & Godsill 1999), and the block sampling particle filter of Doucet et al. (2006). When $k \geq T-1$, we obtain the optimal sequence $\boldsymbol{\psi}^*$. Just as $\boldsymbol{\psi}^*$ cannot typically be used in practice, neither can the exact look-ahead strategies obtained by using (5.6) for some fixed k. In such situations, the proposed look-ahead particle filtering strategies are not $\boldsymbol{\psi}$-APFs, and their relationship to the $\boldsymbol{\psi}^*$-APF is consequently less clear. We note that the offline setting we consider here affords us the freedom to define twisted models using the entire data record $y_{1:T}$. The APF was originally introduced to incorporate a single additional observation, and could therefore be implemented in an online setting, i.e. the algorithm could run while the data record was being produced.

5.2 Asymptotic variance of the $\boldsymbol{\psi}$-APF

Since it is not typically possible to use the sequence $\boldsymbol{\psi}^*$ in practice, we propose to use an approximation of each member of $\boldsymbol{\psi}^*$. In order to motivate such an approximation, we provide a Central Limit Theorem, adapted from the general result of Proposition 1 of Chapter 3. It is convenient to make use of the fact that the estimate $Z_{\boldsymbol{\psi}}^N$ is invariant to rescaling of the functions ψ_t by constants, and we adopt now a particular scaling that simplifies the expression of the asymptotic variance. In particular, we let

$$\bar{\psi}_t(x) := \frac{\psi_t(x)}{\mathbb{E}\left[\psi_t\left(X_t\right) \mid \{Y_{1:t-1} = y_{1:t-1}\}\right]}, \qquad \bar{\psi}_t^*(x) := \frac{\psi_t^*(x)}{\mathbb{E}\left[\psi_t^*\left(X_t\right) \mid \{Y_{1:t-1} = y_{1:t-1}\}\right]},$$

where $\boldsymbol{\psi}^*$ is the optimal sequence of lookahad functions defined in 4.

Proposition 5. *Let $\boldsymbol{\psi}$ be a sequence of bounded, continuous and positive functions. Then*

$$\sqrt{N}\left(\frac{Z_{\boldsymbol{\psi}}^N}{Z} - 1\right) \xrightarrow{d} \mathcal{N}(0, \sigma_{\boldsymbol{\psi}}^2),$$

where,

$$\sigma_{\boldsymbol{\psi}}^2 := \sum_{t=1}^{T} \left\{ \mathbb{E}\left[\frac{\bar{\psi}_t^*\left(X_t\right)}{\bar{\psi}_t\left(X_t\right)} \,\middle|\, \left\{ Y_{1:T} = y_{1:T} \right\} \right] - 1 \right\}. \tag{5.7}$$

Proof of Proposition 5. We define a sequence of densities by

$$\pi_k^{\boldsymbol{\psi}}(x_{1:T}) := \frac{\left[\mu_1^{\boldsymbol{\psi}}\left(x_1\right)\prod_{t=2}^{T} f_t^{\boldsymbol{\psi}}\left(x_{t-1}, x_t\right)\right]\prod_{t=1}^{k} g_t^{\boldsymbol{\psi}}\left(x_t\right)}{\int_{\mathsf{X}^T}\left[\mu_1^{\boldsymbol{\psi}}\left(x_1\right)\prod_{t=2}^{T} f_t^{\boldsymbol{\psi}}\left(x_{t-1}, x_t\right)\right]\prod_{t=1}^{k} g_t^{\boldsymbol{\psi}}\left(x_t\right) dx_{1:T}}, \quad x_{1:T} \in \mathsf{X}^T,$$

for each $k \in \{1, \ldots, T\}$. We also define $\pi_k^{\boldsymbol{\psi}}(x_j) := \int \pi_k(x_{1:j-1}, x_j, x_{j+1:T}) dx_{-j}$ for $j \in \{1, \ldots, T\}$, where $x_{-j} := (x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_N\}$. Combining equation (24.37) of Doucet & Johansen (2011) with elementary manipulations provides,

$$\begin{aligned}
\sigma_{\boldsymbol{\psi}}^2 &= \sum_{t=1}^{T}\left[\int_{\mathsf{X}} \frac{\pi_T^{\boldsymbol{\psi}}(x_t)^2}{\pi_{t-1}^{\boldsymbol{\psi}}(x_t)} dx_t - 1\right] \\
&= \sum_{t=1}^{T}\left[\int_{\mathsf{X}} \frac{\psi_t^*(x_t)}{\psi_t(x_t)} \pi_T^{\boldsymbol{\psi}}(x_t) dx_t \cdot \frac{\int_{\mathsf{X}} \psi_t\left(x_t\right) \pi_{t-1}^{\boldsymbol{\psi}}(x_t) dx_t}{\int_{\mathsf{X}} \psi_t^*\left(x_t\right) \pi_{t-1}^{\boldsymbol{\psi}}(x_t) dx_t} - 1\right] \\
&= \sum_{t=1}^{T}\left\{\mathbb{E}\left[\frac{\psi_t^*\left(X_t\right)}{\psi_t\left(X_t\right)} \middle| \{Y_{1:T} = y_{1:T}\}\right] \frac{\mathbb{E}\left[\psi_t\left(X_t\right) \mid \{Y_{1:t-1} = y_{1:t-1}\}\right]}{\mathbb{E}\left[\psi_t^*\left(X_t\right) \mid \{Y_{1:t-1} = y_{1:t-1}\}\right]} - 1\right\},
\end{aligned}$$

and the expression involving the rescaled terms $\bar{\psi}_t^*$ and $\bar{\psi}_t$ then follows. □

We emphasize that Proposition 5, follows straightforwardly from existing results for Algorithm 3.2, since the $\boldsymbol{\psi}$-APF can be viewed as a BPF for the twisted model defined by $\boldsymbol{\psi}$. For example, in the case $\boldsymbol{\psi}$ consists only of constant functions, we obtain the standard asymptotic variance for the BPF

$$\sigma^2 = \sum_{t=1}^{T} \left\{ \mathbb{E}\left[\bar{\psi}_t^* \left(X_t\right) \mid \left\{Y_{1:T} = y_{1:T}\right\} \right] - 1 \right\}.$$

From Proposition 5 we can deduce that $\sigma_{\boldsymbol{\psi}}^2$ tends to 0 as $\boldsymbol{\psi}$ approaches $\boldsymbol{\psi}^*$ in an appropriate sense. Hence, Propositions 4 and Proposition 5 together provide some justification for designing particle filters by approximating the sequence $\boldsymbol{\psi}^*$.

Chapter 6

The general iAPF algorithm

6.1 Classes of f and ψ

While the $\boldsymbol{\psi}$-APF described in Section 5.1 and the asymptotic results just described are valid very generally, practical implementation of the $\boldsymbol{\psi}$-APF does impose some restrictions jointly on the transition densities f and functions in $\boldsymbol{\psi}$. For tractability here we consider only the case where the HMM's initial distribution is a mixture of Gaussians and f is a member of $\mathcal{F}$, the class of transition densities of the form

$$f\left(x,\cdot\right)=\sum_{k=1}^{M}c_k(x)\mathcal{N}\left(\,\cdot\,;a_k\left(x\right),b_k\left(x\right)\right), \tag{6.1}$$

where $M\in\mathbb{N}$, and $(a_k)_{k\in\{1,\ldots,M\}}$ and $(b_k)_{k\in\{1,\ldots,M\}}$ are sequences of mean and covariance functions, respectively and $(c_k)_{k\in\{1,\ldots,M\}}$ a sequence of $\mathbb{R}_+$-valued functions with $\sum_{k=1}^{M}c_k(x)=1$ for all $x\in\mathsf{X}$. Let Ψ define the class of functions of the form

$$\psi(x)=C+\sum_{k=1}^{M}c_k\mathcal{N}\left(x;a_k,b_k\right), \tag{6.2}$$

where $M\in\mathbb{N}$, $C\in\mathbb{R}_+$, and $(a_k)_{k\in\{1,\ldots,M\}}$, $(b_k)_{k\in\{1,\ldots,M\}}$ and $(c_k)_{k\in\{1,\ldots,M\}}$ are a sequence of means, covariances and positive real numbers, respectively. As the function ψ appears at the denominator of the incremental weights, adding the constant C is functional to the robustness of the algorithm as it guarantees the weights are bounded and with non explosive variance. Note that a product of Gaussian densities is itself a Gaussian density up to a multiplying constant whose mean and variance/covariance matrix are easily derived (see for example Bromiley (2003)). Consequently also the product of two mixtures of Gaussian densities is (proportional to) a mixture of Gaussian densities. When $f \in \mathcal{F}$ we can choose $\psi_t \in \Psi$, so that it is straightforward to implement Algorithm 5.1 since, for each $t \in \{1, \ldots, T\}$, both $\psi_t(x)$ and $\tilde{\psi}_{t-1}(x) = f(x, \psi_t)$ can be computed explicitly and $f_t^{\boldsymbol{\psi}}(x, \cdot)$ is a mixture of normal distributions whose component means and covariance matrices can also be computed. In this case the constant C also guarantees that $f_t^{\psi}(x, \cdot)$ is a mixture of densities with some non-zero wright associated with the mixture component $f(x, \cdot)$.

6.2 Recursive approximation of $\boldsymbol{\psi}^*$

The ability to compute $f(\cdot, \psi_t)$ pointwise when $f \in \mathcal{F}$ and $\psi_t \in \Psi$ is also instrumental in the recursive function approximation scheme we now describe. Our approach is based on the following observation.

Proposition 6. *The sequence $\boldsymbol{\psi}^*$ satisfies $\psi_T^*(x_T) = g(x_T, y_T)$, $x_T \in \mathsf{X}$ and*

$$\psi_t^*(x_t) = g(x_t, y_t) f\left(x_t, \psi_{t+1}^*\right), \quad x_t \in \mathsf{X}, \quad t \in \{1, \ldots, T-1\}. \tag{6.3}$$

Proof. The definition of $\boldsymbol{\psi}^*$ provides that $\psi_T^*(x_T) = g(x_T, y_T)$. For $t \in \{1, \ldots, T-1\}$,

$$\begin{aligned} & g(x_t, y_t) f\left(x_t, \psi_{t+1}^*\right) \\ = \;& g(x_t, y_t) \int_{\mathsf{X}} f(x_t, x_{t+1}) \mathbb{E}\left[\prod_{p=t+1}^{T} g(X_p, y_p) \mid \{X_{t+1} = x_{t+1}\}\right] dx_{t+1} \\ = \;& g(x_t, y_t) \mathbb{E}\left[\prod_{p=t+1}^{T} g(X_p, y_p) \mid \{X_t = x_t\}\right] \\ = \;& \psi_t^*(x_t). \quad \square \end{aligned}$$

Let $(\xi_1^{1:N}, \ldots, \xi_T^{1:N})$ be random variables obtained by running a particle filter. We propose to approximate $\boldsymbol{\psi}^*$ by Algorithm 6.1, for which we define $\psi_{T+1} \equiv 1$. This algorithm mirrors the backward sweep of the forward filtering backward smoothing recursion (see for instance Doucet et al. (2000)) which, if it could be calculated, would yield exactly $\boldsymbol{\psi}^*$.

Algorithm 6.1 Recursive function approximations

```
For $t = T, \ldots, 1$:
    1. Set $\psi_t^i \leftarrow g\left(\xi_t^i, y_t\right) f\left(\xi_t^i, \psi_{t+1}\right)$ for $i \in \{1, \ldots, N\}$.
    2. Choose $\psi_t$ as a member of $\Psi$ on the basis of $\xi_t^{1:N}$ and $\psi_t^{1:N}$.
```

One choice in step 2. of Algorithm 6.1 is to define ψ_t using a non-parametric approximation such as a Nadaraya–Watson estimate (Nadaraya 1964, Watson 1964). An alternative approach is to choose ψ_t as the minimizer in some subset of Ψ of some function of ψ_t, $\xi_t^{1:N}$ and $\psi_t^{1:N}$. Although a number of other choices are possible, we focus in Chapter 7 on these two approaches.

6.3 The algorithm

The iterated auxiliary particle filter (iAPF), Algorithm 6.2, is obtained by iteratively running a $\boldsymbol{\psi}$-APF and estimating $\boldsymbol{\psi}^*$ from its output. Specifically, after each $\boldsymbol{\psi}$-APF is run, $\boldsymbol{\psi}^*$ is re-approximated using the particles obtained, and the number of particles may be increased according to a well-defined rule. The algorithm terminates when a stopping rule is satisfied.

Algorithm 6.2 An iterated auxiliary particle filter with parameters (N_0, k, τ)

```
1. Initialize: set $\boldsymbol{\psi}^0$ to be a sequence of constant functions, $l \leftarrow 0$.
2. Repeat:
    (a) Run a $\boldsymbol{\psi}^l$-APF with $N_l$ particles, and set $\hat{Z}_l \leftarrow Z_{\boldsymbol{\psi}^l}^{N_l}$.
    (b) If $l > k$ and $\mathrm{sd}(\hat{Z}_{l-k:l})/\mathrm{mean}(\hat{Z}_{l-k:l}) < \tau$, go to 3.
    (c) Compute $\boldsymbol{\psi}^{l+1}$ using a version of Algorithm 6.1 with the particles
        produced.
    (d) If $N_{l-k} = N_l$ and the sequence $\hat{Z}_{l-k:l}$ is not monotonically increasing,
        set $N_{l+1} \leftarrow 2N_l$. Otherwise, set $N_{l+1} \leftarrow N_l$.
    (e) Set $l \leftarrow l + 1$ and go back to 2a.
3. Run a $\boldsymbol{\psi}^l$-APF and return $\hat{Z} := Z_{\boldsymbol{\psi}}^{N_l}$
```

Note that Step 3 of Algorithm 6.2 is of vital importance. If we do not run one more iteration of the iAPF after the stopping criterion is met, the unbiasedness of the normalising constant estimates is compromised. The rationale for step 2(d) of Algorithm 6.2 is that if the sequence $\hat{Z}_{l-k:l}$ is monotonically increasing, there is some evidence that the approximations $\boldsymbol{\psi}^{l-k:l}$ are improving, and so increasing the number of particles may unnecessarily increase computational cost. However, if the approximations $\hat{Z}_{l-k:l}$ have both high relative standard deviation in comparison to τ and are oscillating then reducing the variance of the approximation of Z and/or improving the approximation of $\boldsymbol{\psi}^*$ may require an increased number of particles. Some support for this procedure can be obtained from the log-normal CLT of Bérard et al. (2014): under regularity assumptions, $\log Z_{\boldsymbol{\psi}}^N$ is approximately a $\mathcal{N}(-\delta_{\boldsymbol{\psi}}^2/2, \delta_{\boldsymbol{\psi}}^2)$ random variable and so $\mathbb{P}\left(Z_{\boldsymbol{\psi}'}^N \geq Z_{\boldsymbol{\psi}}^N\right) \approx 1 - \Phi\left(\left[\delta_{\boldsymbol{\psi}'}^2 - \delta_{\boldsymbol{\psi}}^2\right] / \left[2\sqrt{\delta_{\boldsymbol{\psi}}^2 + \delta_{\boldsymbol{\psi}'}^2}\right]\right)$, which is close to 1 when $\delta_{\boldsymbol{\psi}'}^2 \ll \delta_{\boldsymbol{\psi}}^2$ and provided that δ_ψ is $\mathcal{O}(1)$ (which always the case for our examples and applications). Note that throughout all the simulations we used the diagnostic $\mathrm{sd}(\hat{Z}_{l-k:l})/\mathrm{mean}(\hat{Z}_{l-k:l})$ for Step (b) of Algorithm 6.2, because of good empirical results on preliminary simulations. We point out the suggestion that a more natural quantity would be $\mathrm{sd}\left(\log \hat{Z}_{l-k:l}\right)$ since optimality results in the literature are generally phrased in terms of this.

6.4 Approximations of smoothing expectations

Thus far, we have focused on approximations of the marginal likelihood, L, associated with a particular model and data record $y_{1:T}$. Particle filters are also used to approximate so-called smoothing expectations, i.e. quantities of the type $\pi(\varphi) := \mathbb{E}\left[\varphi(X_{1:T}) \mid \{Y_{1:T} = y_{1:T}\}\right]$ for some $\varphi : \mathsf{X}^{\mathsf{T}} \to \mathbb{R}$. Such approximations can be motivated by a slight extension of,

$$\gamma(\varphi) := \int_{\mathsf{X}^T} \varphi(x_{1:T})\mu_1\left(x_1\right) g_1\left(x_1\right) \prod_{t=2}^{T} f_t\left(x_{t-1}, x_t\right) g_t\left(x_t\right) dx_{1:T},$$

where φ is a real-valued, bounded, continuous function. We can write $\pi(\varphi) = \gamma(\varphi)/\gamma(1)$, where 1 denotes the constant function $x \mapsto 1$. We define below a well-known, unbiased and strongly consistent estimate $\gamma^N(\varphi)$ of $\gamma(\varphi)$, which can be obtained from Algorithm 3.2. A strongly consistent approximation of $\pi(\varphi)$ can then be defined as $\gamma^N(\varphi)/\gamma^N(1)$.

The definition of $\gamma^N(\varphi)$ is facilitated by a specific implementation of step 2. of Algorithm 3.2 in which one samples

$$A_{t-1}^i \sim \text{Categorical}\left(\frac{g_{t-1}(\xi_{t-1}^1)}{\sum_{j=1}^N g_{t-1}(\xi_{t-1}^j)}, \ldots, \frac{g_{t-1}(\xi_{t-1}^N)}{\sum_{j=1}^N g_{t-1}(\xi_{t-1}^j)}\right), \qquad \xi_t^i \sim f_t(\xi_{t-1}^{A_{t-1}^i}, \cdot),$$

for each $i \in \{1, \ldots, N\}$ independently. Other resampling schemes could also be used and are consistent with the "ancestor selection" view. Use of, e.g., the Alias algorithm (Walker 1974, 1977) gives the algorithm $\mathcal{O}(N)$ computational complexity, and the random variables $(A_t^i; t \in \{1, \ldots, T-1\}, i \in \{1, \ldots, N\})$ provide ancestral information associated with each particle. By defining recursively for each $i \in \{1, \ldots, N\}$, $B_T^i := i$ and $B_{t-1}^i := A_{t-1}^{B_t^i}$ for $t = T, \ldots, 2$, the $\{1, \ldots, N\}^T$-valued random variable $B_{1:T}^i$ encodes the ancestral lineage of ξ_T^i (Andrieu et al. 2010). It follows from Del Moral (2004, Theorem 7.4.2) that the approximation

$$\gamma^N(\varphi) := \left[\frac{1}{N}\sum_{i=1}^N g_T(\xi_T^i)\varphi(\xi_1^{B_1^i}, \xi_2^{B_2^i}, \ldots, \xi_T^{B_T^i})\right] \prod_{t=1}^{T-1}\left(\frac{1}{N}\sum_{i=1}^N g_t(\xi_t^i)\right),$$

is unbiased and strongly consistent, and a strongly consistent approximation of

$\pi(\varphi)$ is

$$\pi^N(\varphi) := \frac{\gamma^N(\varphi)}{\gamma^N(1)} = \frac{1}{\sum_{i=1}^N g_T(\xi_T^i)} \sum_{i=1}^N \varphi\left(\xi_1^{B_1^i}, \xi_2^{B_2^i}, \ldots, \xi_T^{B_T^i}\right) g_T(\xi_T^i). \qquad (6.4)$$

The $\boldsymbol{\psi}^*$-APF is optimal in terms of approximating $\gamma(1) \equiv Z$ and not $\pi(\varphi)$ for general φ. Asymptotic variance expressions akin to Proposition 5, but for $\pi_{\boldsymbol{\psi}}^N(\varphi)$, can be derived using existing results (see, e.g., Del Moral & Guionnet 1999, Chopin 2004, Künsch 2005, Douc & Moulines 2008) in the same manner. These could be used to investigate the influence of $\boldsymbol{\psi}$ on the accuracy of $\pi_{\boldsymbol{\psi}}^N(\varphi)$ or the interaction between φ and the sequence $\boldsymbol{\psi}$ which minimizes the asymptotic variance of the estimator of its expectation.

Finally, we observe that when the optimal sequence $\boldsymbol{\psi}^*$ is used in an APF in conjunction with an adaptive resampling strategy (see Algorithm 7.1 below), the weights are all equal, no resampling occurs and the ξ_t^i are all i.i.d. samples from $\mathbb{P}(X_t \in \cdot \mid \{Y_{1:T} = y_{1:T}\})$. This at least partially justifies the use of iterated $\boldsymbol{\psi}$-APFs to approximate $\boldsymbol{\psi}^*$: the asymptotic variance $\sigma_{\boldsymbol{\psi}}^2$ in (5.7) is particularly affected by discrepancies between $\boldsymbol{\psi}^*$ and $\boldsymbol{\psi}$ in regions of relatively high conditional probability given the data record $y_{1:T}$, which is why we have chosen to use the particles as support points to define approximations of $\boldsymbol{\psi}^*$ in Algorithm 6.1.

Chapter 7

Implementation Details and Recursive Function Approximation Approaches

In this section we provide all the details that are necessary for the implementation of the iAPF. In particular we specify two possible approaches for the backward function approximation in Algorithm 6.1 that make use of the statistical tools of kernel density estimates (in Section 7.2) and parametric optimisation (in Section 7.3).

7.1 Implementation details

In Algorithm 6.2 we need to specify the parameters (N_0, k, τ). The choice of N_0 is not critical, as the iAPF scheme increases adaptively the number of particles until the desired degree of precision is reached. We specify the number N_0 for each of our examples and applications. For the stopping rule, we used $k = 5$ for the application in Chapter 8, and $k = 3$ for the applications in Chapter 9. The parameter value $k = 5$ was used initially but it seemed overly conservative for our applications, where $k = 3$ proved empirically to be reliable enough. In cases where a preliminary study brings some concern about the convergence of the look-ahead functions an higher value of k is advisable. We observed empirically that the relative standard deviation of the likelihood estimate tended to be close to, and often smaller than, the chosen level for τ. A value of $\tau = 1$ should therefore be sufficient to keep the relative standard deviation around 1 as desired (see, e.g., Doucet et al. 2015, Sherlock et al. 2015). We set $\tau = 0.5$ as a conservative choice for all our simulations apart from the multivariate stochastic volatility model of Section 9.2, where we set $\tau = 1$ to improve speed.

We used an effective sample size-based resampling scheme (Kong et al. 1994, Liu & Chen 1995), described in Algorithm 7.1 with a user-specified parameter $\kappa \in [0, 1]$. The effective sample size is defined as $\mathrm{ESS}(W^1, \dots, W^N) := \left(\sum_{i=1}^N W^i\right)^2 / \sum_{i=1}^N (W^i)^2$, and the estimate of Z is

$$Z^N := \prod_{t \in \mathcal{R} \cup \{T\}} \left[\frac{1}{N} \sum_{i=1}^N W_t^i \right],$$

where $\mathcal{R}$ is the set of "resampling times" defined as

$$\mathcal{R} := \left\{ t \in \{1, \dots, T-1\} : \mathrm{ESS}(W_t^1, \dots, W_t^N) \le \kappa N \right\}.$$

This reduces to Algorithm 5.1 when $\kappa = 1$ and to a simple importance sampling algorithm when $\kappa = 0$; we use $\kappa = 0.5$ in our simulations. The use of adaptive resampling is motivated by the fact that when the effective sample size is large, resampling can be detrimental in terms of the quality of the approximation Z^N.

7.2 Kernel density estimate approach

We present a first implementation that combines the iAPF scheme with the statistical tool of kernel density estimation to incorporate the information acquired into the look-ahead function update step at the end of every iteration.

Kernel density estimation is a non-parametric technique to estimate the probability density function of a random variable. Let $\{\xi^i, w^i\}_{i \in 1:N}$ be a weighted sample drawn from some distribution with unknown density f. We are interested in estimating the shape of this function f. Its kernel density estimator is

$$\hat{f}_h(x) = \Phi_h \left(\{\xi^i, w^i\}_{i \in 1:N} \right)(x) := \sum_{i=1}^N w^i \cdot K_h \left(x - \xi^i \right)$$

Algorithm 7.1 $\boldsymbol{\psi}$-Auxiliary Particle Filter with κ-adaptive resampling

```
1. Sample $\xi_1^i \sim \mu_1^{\boldsymbol{\psi}}$ independently, and set $W_1^i \leftarrow g_1^{\boldsymbol{\psi}}(\xi_1^i)$ for $i \in \{1, \ldots, N\}$.
2. For $t = 2, \ldots, T$:
    (a) If $\mathrm{ESS}(W_{t-1}^1, \ldots, W_{t-1}^N) \leq \kappa N$, sample independently
            $$\xi_t^i \sim \frac{\sum_{j=1}^N W_{t-1}^j f_t^{\boldsymbol{\psi}}(\xi_{t-1}^j, \cdot)}{\sum_{j=1}^N W_{t-1}^j}, \qquad i \in \{1, \ldots, N\},$$
        and set $W_t^i \leftarrow g_t^{\boldsymbol{\psi}}(\xi_t^i)$, $i \in \{1, \ldots, N\}$.
    (b) Otherwise, sample $\xi_t^i \sim f_t^{\boldsymbol{\psi}}(\xi_{t-1}^i, \cdot)$ independently, and set $W_t^i \leftarrow W_{t-1}^i g_t^{\boldsymbol{\psi}}(\xi_t^i)$ for $i \in \{1, \ldots, N\}$.
```

where the kernel K_h is a symmetric function that integrates to one and h is a smoothing parameter called the bandwidth (possibly a matrix for multidimensional models). Generally one wants to choose h as small as the cardinality of the set $\{\xi^i, w^i\}_{i \in 1:N}$ allows, however there is always a trade-off between the bias of the function estimator $\hat{f}_h$ and its smoothness. For a detailed exposition of kernel density estimation see Rosenblatt et al. (1956), Parzen (1962) and for weighted kernel density estimation and bandwidth selection see Chiu (1991), Loader (1999), Wang & Wang (2007).

We want to produce a kernel density function that approximates at least locally (in a region of high probability with respect to the smoothing distribution) the optimal look-ahead function ψ_t^* up to a constant of proportionality and for all $t \in 1:T$. To do so we combine the tools of importance sampling and weighted kernel density estimation. Ideally we would like to be able to draw a sample $\xi_t^{1:M}$ from a distribution with density q_t with high probability mass in the region of interest and assign to each particle ξ_t^i a weight w_t^i proportional to $\psi_t^*\left(\xi_t^i\right)/q_t\left(\xi_t^i\right)$, for each $t \in 1:N$. We could then define ψ_t as

$$\psi_t\left(x\right) = \Phi_h\left(\left\{\xi_t^i, w_t^i\right\}_{i \in 1:M}\right)\left(x\right) + C,$$

with the reason for adding the constant C is given in Section 6.1. Even if we could define a suitable proposal q_t, in general we have no access to the optimal look-ahead function $\boldsymbol{\psi}^*$. We can exploit instead the filtering support ξ_t^i at time t (here ξ_t^i refers to a realisation of the particle dynamic rather than the stochastic system of particles itself) and the approximations ψ_t^i defined through the backwards procedure in Algorithm 6.1. Note that $\xi_t^{1:N}$ is a conditional independent sample given the set of particles and weights $\left\{\xi_{t-1}^{1:N}, w_{t-1}^{1:N}\right\}$: we call the conditional distribution density of its element q_t. This is neither the smoothing nor the filtering distribution density, as we are not taking into account the filtering weights. We want to assign importance weights to the sample $\xi_t^{1:N}$. In order to correct for the distribution the sample $\xi_t^{1:N}$ is drawn from without having access to the density q_t, we define a kernel density approximation of q_t as

$$\hat{q}_t\left(x\right) = \Phi_h\left(\left\{\xi_t^i, \frac{1}{N}\right\}_{i\in 1:N}\right)\left(x\right).$$

For the term at the numerator of the importance weights we use for every particle ξ_t^i the approximation ψ_t^i of $\psi_t^*\left(\xi_t^i\right)$ we obtained through the backward recursive procedure as in algorithm 6.1. Using these ingredients we define the look-ahead function ψ_t at time t as

$$\psi_t\left(x\right) = \Phi_h\left(\xi_t^i, v_t^i\right)\left(x\right) + C$$

where the kernel density estimate weights $v_t^{1:N}$ are given by

$$v_t^i = \frac{\psi_t^i}{\hat{q}_t\left(\xi_t^i\right)}$$

for all $i \in 1:N$. The kernel density estimate implementation of Algorithm 6.1 corresponds to Algorithm 7.2, with the initialisation $\psi_{T+1} \equiv 1$.

Algorithm 7.2 Recursive function approximations, kernel density estimate approach.

```
For $t = T, \dots, 1$:
    1. Set $\psi_t^i \leftarrow g\left(\xi_t^i, y_t\right) f\left(\xi_t^i, \psi_{t+1}\right)$ for $i \in \{1, \dots, N\}$.
    2. Define $\hat{q}_t(x) = \Phi_h\left(\left\{\xi_t^i, \frac{1}{N}\right\}_{i\in 1:N}\right)(x)$.
        (a) Set $v_t^i \leftarrow \frac{\psi_t^i}{\hat{q}_t\left(\xi_t^i\right)}$ for $i \in \{1, \dots, N\}$.
        (b) Set $\Psi \ni \psi_t(x) = \Phi_h\left(\xi_t^i, v_t^i\right)(x) + C$
```

We choose the kernel K_{h_t} to be the density function of a (possibly multivariate) Normal distribution $\mathcal{N}(\underline{0}, h_t H)$

$$K_{h_t}\left(x - \xi_t^{1:N}\right) := \mathcal{N}\left(x; \xi_t^i, h_t H\right)$$

where the bandwidth parameter h_t and the matrix H depend on the set of points $\xi_t^{1:N}$, and we define ψ_t as an appropriately weighted sum of Gaussian densities

$$\psi_t(x) = \sum_{i\in 1:N} v_t^i \mathcal{N}\left(x; x_t^i, h_t H\right) + C,$$

for all $t \in 1 : T$. This guarantees that ψ_t belongs to the chosen class Ψ_θ of look-ahead functions defined in subsection 6.1. In general one would choose kernel density functions which are conjugate with respect to the model transition in order to be able to perform easily the backward recursion procedure. We use Scott's rule (see Silverman (1982)) for the bandwidth choice, allowing just diagonal covariance matrices in multidimensional models, due to algorithm efficiency and because we find that the accuracy gain from a more refined estimate is negligible.

For a fixed $t \in 1 : T$, the kernel density estimate $\hat{q}_t$ depends also on the random set of weighted particles $\left\{\xi_{t-1}^{1:N}, w_{t-1}^{1:N}\right\}$, and the behaviour of the set $\psi_t^{1:N}$ is even less clear, therefore it is difficult to analyse the asymptotic behaviour of the look-ahead function ψ_t as the number of particles N increases. Empirical results suggest that the number of particles used is critical in getting a good estimate of the optimal look-ahead function sequence $\boldsymbol{\psi}^*$, meaning that as $N \longrightarrow \infty$

then ψ_t approaches ψ_t^* in a suitable sense. Nonetheless they also suggest that at least in this context an extremely accurate estimate of $\boldsymbol{\psi}^*$ is *not* critical in the performance of the iAPF algorithm: the idea is that even if the look-ahead function approximation is not very accurate, it can efficiently guide the particles through regions of the path space with higher probability mass. Increasing the number of particles N comes with a greatly increased computational cost which is of order $\mathcal{O}\left(N^2\right)$ for the kernel density estimate implementation. In order to reduce the computational burden we can use for the backward recursion only a subset of the support particles. A natural choice is to sample without replacement from the set $\xi_t^{1:N}$ the desired number of particles, so that we still have a sample of elements drawn independently from a conditional distribution. However for our simulations we find empirically that is more efficient to select for each time step $t \in 1:T$ a subset $\xi_t^{J_t}$ of the particles $\xi_t^{1:N}$ where the cardinality of the subset of indices J_t is $\mid J_t \mid= \sqrt{N}$, and these are chosen picking the particles corresponding to the $\sqrt{N}$ highest values ψ_t^J. For our simulations a set of $\sqrt{N}$ is sufficient to obtain good kernel density estimates of the look-ahead functions and this way we achieve the more affordable computational cost of order $\mathcal{O}\left(N\sqrt{N}\right)$.

7.3 Parametric approach

When we use kernel density estimates, for every $t \in 1:T$ the look-ahead function ψ_t takes the form of a weighted sum of the chosen kernels (plus a constant). Increasing the number of kernels in the mixture, we could approximate arbitrarily well the optimal look-ahead function ψ_t^*. This comes at a prohibitive computational cost. As we mention in the previous subsection, an extremely accurate estimate of ψ_t^* is not crucial: using a reduced number of kernels speeds up the algorithm significantly, without necessarily compromising the efficiency. Taking this to the extreme and we could reduce to one the number of components in the mixture.

We present a second implementation that defines ψ_t as a density function chosen from a class of density functions Ψ_θ through parameter optimisation. We choose Ψ_θ to be the class of normal density functions, so that $\theta = (m, \Sigma)$ represents the mean and covariance matrix parameters. Specifically, for each

$t \in \{1, \ldots, T\}$, we compute numerically

$$(m_t^*, \Sigma_t^*, \lambda_t^*) = \text{argmin}_{(m,\Sigma,\lambda)} \sum_{i=1}^{N} \left[\mathcal{N}\left(\xi_t^i; m, \Sigma\right) - \lambda \psi_t^i\right]^2 + l\left(\xi_t^{1:N}, N, m, \Sigma, \lambda\right), \tag{7.1}$$

where $m \in \mathbb{R}^d$, $\Sigma \in \mathbb{R}^{d\times d}$, $\lambda \in \mathbb{R}$ and l is an appropriate real function that penalises values of m too distant in an appropriate sense from the points $\xi_t^{1:N}$. We then set

$$\psi_t(x_t) := \mathcal{N}\left(x_t; m_t^*, \Sigma_t^*\right) + c(N, m_t^*, \Sigma_t^*), \tag{7.2}$$

where c is a positive real-valued function, which ensures that $f_t^{\boldsymbol{\psi}}(x, \cdot)$ is a mixture of densities with some non-zero weight associated with the mixture component $f(x, \cdot)$. This is intended to guard against terms in the asymptotic variance $\sigma_{\boldsymbol{\psi}}^2$ in (5.7) being very large or unbounded. We chose (7.1), coupled with a local optimiser, for simplicity and its low computational cost, and it provided good performance in our simulations. Different objective functions can be considered. To justify the introduction of the term $l\left(\xi_t^{1:N}, N, m, \Sigma, \lambda\right)$ note that

$$\lim_{\substack{|m| \to \infty \\ \lambda \to 0}} \sum_{i=1}^{N} \left[\mathcal{N}\left(\xi_t^i; m, \Sigma\right) - \lambda \psi_t^i\right]^2 = 0$$

independently from $\xi_t^{1:N}$ and Σ. In this case an unconstrained optimiser can return as minimising triplet (m, Σ, λ) the trivial values of $\lambda = 0$, a mean m extremely far from the origin (very high value of $|m|$) and a covariance matrix Σ with very small diagonal entries. To address this problem in general we can introduce a Tikhonov regularisation term $l\left(\xi_t^{1:N}, N, m, \Sigma, \lambda\right)$ in the objective function, which consists of a norm penalty on the optimisation arguments (m, Σ, λ). This is a popular technique in statistics and especially in the field of inverse problems (see e.g. Tikhonov et al. (1995)). The introduction of the term $l\left(\xi_t^{1:N}, N, m, \Sigma, \lambda\right)$ in the objective function is not necessary for all the simulations in Part 3, as an implicit form of regularisation is given by the adoption of a local optimiser. In particular we used the general-purpose optimiser R function `optim` based on the `BFGS` quasi-Newton method. In all of the examples and applications of Part III, the chosen optimiser proved to be effective in finding the local optima corresponding to local non-trivial solution of the minimisation problem 7.1 corresponding to $l \equiv 0$. For general problems where there is some concern about the optimiser returning trivial solutions, we point out the method `L-BFGS-B`, also relative to the `optim` function, which consists of a modification of the `BFGS` quasi-Newton method that allows for box constraint optimisation. We do include the Tikhonov regularisation term for the simulations with diffusion processes in Part IV, where an appropriate penalising function l is defined. We performed the minimization in (7.1) under the restriction that Σ was a diagonal matrix, as this was considerably faster and preliminary simulations suggested that this was adequate for the examples considered.

Algorithm 7.3 Recursive function approximations, parametric approach.

```
For $t = T, \dots, 1$:
    1. Set $\psi_t^i \leftarrow g\left(\xi_t^i, y_t\right) f\left(\xi_t^i, \psi_{t+1}\right)$ for $i \in \{1, \dots, N\}$.
    2. Compute numerically
       $$\left(m_t^*, \Sigma_t^*, \lambda_t^*\right) = \text{argmin}_{(m,\Sigma,\lambda)} \sum_{i=1}^{N} \left[\mathcal{N}\left(\xi_t^i; m, \Sigma\right) - \lambda\psi_t^i\right]^2 + l\left(\xi_t^{1:N}, N, m, \Sigma, \lambda\right),$$
       and set $\Psi \ni \psi_t(x) = \mathcal{N}\left(x; m_t^*, \Sigma_t^*\right) + C$
```

7.4 A note on the simulations

The purpose of the next part of the thesis is to investigate the performance of the iAPF in a set of examples and applications. We show how in interesting scenarios the iAPF can provide substantially better estimates of the marginal likelihood L than the BPF at the same computational cost. This is exemplified by its performance in estimating the marginal likelihood L of extreme observations and when d is large, recalling that $\mathsf{X} = \mathbb{R}^d$. In these cases, the BPF typically requires a large number of particles in order to approximate L accurately. In contrast, the $\boldsymbol{\psi}^*$-APF computes L exactly, and we investigate below the extent to which the iAPF is able to provide accurate approximations in this setting. Similarly, when there are unknown statistical parameters θ, we show empirically that the accuracy of iAPF approximations of the likelihood $L(\theta)$ are more robust to changes in θ than their BPF counterparts.

Unbiased, non-negative approximations of likelihoods $L(\theta)$ are central to the particle marginal Metropolis–Hastings algorithm (PMMH) of Andrieu et al. (2010), a prominent parameter estimation algorithm for general state space hidden Markov models. An instance of a pseudo-marginal Markov chain Monte Carlo algorithm (Beaumont 2003, Andrieu & Roberts 2009), the computational efficiency of PMMH depends, sometimes dramatically, on the quality of the unbiased approximations of $L(\theta)$ (Andrieu & Vihola 2015, Lee & Łatuszyński 2014, Sherlock et al. 2015, Doucet et al. 2015) delivered by a particle filter for a range of θ values. The relative robustness of iAPF approximations of $L(\theta)$ to changes in θ, mentioned above, motivates their use over BPF approximations in PMMH.

Part III

Examples and Applications to State Space Models

Chapter 8

The Linear Gaussian Model

Recall from Section 2.2.1 that a linear Gaussian HMM is defined by the following initial, transition and observation Gaussian densities: $\mu(\cdot) = \mathcal{N}(\cdot; m, \Sigma)$, $f(x, \cdot) = \mathcal{N}(\cdot; Ax, B)$ and $g(x, \cdot) = \mathcal{N}(\cdot; Cx, D)$, where $m \in \mathbb{R}^d$, $\Sigma, A, B \in \mathbb{R}^{d\times d}$, $C \in \mathbb{R}^{d\times d'}$ and $D \in \mathbb{R}^{d'\times d'}$. For this model, it is possible to implement the fully adapted APF (FA-APF) and to compute explicitly the marginal likelihood, filtering and smoothing distributions using the Kalman filter, facilitating comparisons. We emphasize that implementation of the FA-APF is possible only for a restricted class of analytically tractable models, while the iAPF methodology is applicable more generally. Nevertheless, the iAPF exhibited better performance than the FA-APF in our examples.

8.1 Exploratory simulations with the kernel density estimates approach

Using kernel density estimates as described in 7.2 is a natural first approach to shape the sequence of look-ahead functions $\boldsymbol{\psi}$ as with this method we do not need any a priori knowledge on $\boldsymbol{\psi}^*$ (e.g. if the perfect look-ahead functions belong to a specific class of functions). In this section when the stopping rule in Algorithm 6.2 is met we keep running the algorithm for some more iterations. We do so to show that iterating the iAPF further leads to little to no gain in the accuracy of the estimates, which supports empirically our choice of stopping rule.

8.1.1 Simple LG model

We sample a realisation $x_{1:T}$ of the latent process up to time T and a realisation $y_{1:T}$ of the observation process given $x_{1:T}$ for different configurations of parameters. We look at subsequent estimates of the sequence of look-ahead functions $\boldsymbol{\psi}^l$ in a 2-dimensional linear Gaussian model, with the set of parameters

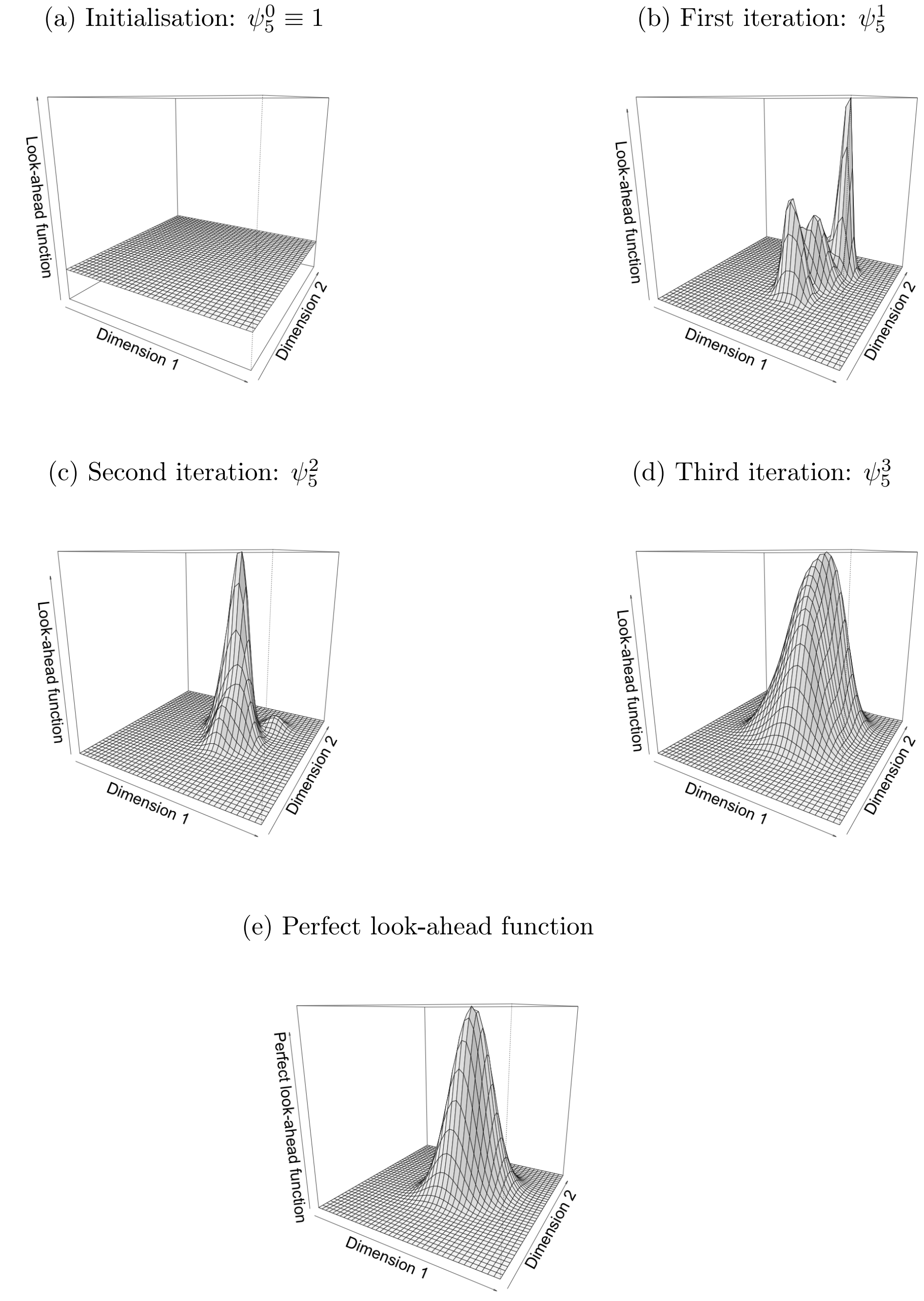
$$m = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \Sigma = A = B = C = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, D = \begin{pmatrix} 0.5 & 0.2 \\ 0.2 & 0.5 \end{pmatrix}$$

where we set $T = 10$. We chose this configuration for simplicity as the fact that the latent process is explosive is not a problem given the very short observation sequence. Recall that at every iteration of the algorithm, a new sequence of look-ahead functions $\boldsymbol{\psi}^l$ is defined, until the stopping rule in 6.2 is met. We look here at the first three iterations of the algorithm and we report in Figure 8.1 the fifth function ψ_5^l of the sequence $\boldsymbol{\psi}^l$ for $l \in 0 : 3$ along with the perfect look-ahead function, i.e. a multivariate Gaussian density function whose parameters can be determined by Kalman filtering. The look-ahead function ψ_5^0 at the first iteration is flat, as we assumed no a priori information is available, but after the third iteration it is already difficult to distinguish the approximation from the perfect look-ahead function, the last entry in the table.

A more rigorous analysis can be done by comparing the empirical variance of the normalising constant estimator $\hat{Z}$ and the average resampling count for different number of total algorithm iterations l.

Table 1 reports the empirical variance over fifty runs of the normalising constant estimator for the iterative look-ahead SMC algorithm at different iteration steps, including the bootstrap case with $l = 0$, and $N = 500$ particles. The first thing to notice is the substantial improvement of the accuracy of the estimator even with low values of l compared to the Bootstrap particle filter: after only two iterations, the variance of the estimator is reduced about tenfold. The second thing we notice is that the increments in efficiency given by further iterating the algorithm quickly become negligible, so that after three iterations no appreciable improvement is achieved.

Figure 8.1: Subsequent look-ahead function estimates for ψ_5^l with $l = 0, 1, 2, 3$. Comparison with perfect look-ahead function.



(a) Initialisation: $\psi_5^0 \equiv 1$

(b) First iteration: ψ_5^1

(c) Second iteration: ψ_5^2

(d) Third iteration: ψ_5^3

(e) Perfect look-ahead function

Table 8.1: Empirical variance of the estimator of the logarithm of the normalising constant at different iteration count l

	Variance of the loglikelihood estimator
Bootstrap	$3.4 \cdot 10^{-2}$
l=1	$1.1 \cdot 10^{-2}$
l=2	$2.8 \cdot 10^{-3}$
l=3	$1.2 \cdot 10^{-3}$
l=5	$1.2 \cdot 10^{-3}$
l=10	$9.6 \cdot 10^{-4}$
l =20	$1.0 \cdot 10^{-3}$

We report in Table 2 the count of total resampling count over the time horizon $T = 10$ with different values of l, averaged over fifty runs of the algorithm.

Table 8.2: Averages over 50 runs of the iterative look-ahead algorithm

Average Resampling Count						
Bootstrap	l=1	l=2	l=3	l=5	l=10	l=20
10	4.64	1.76	1.36	1.16	1.28	1.30

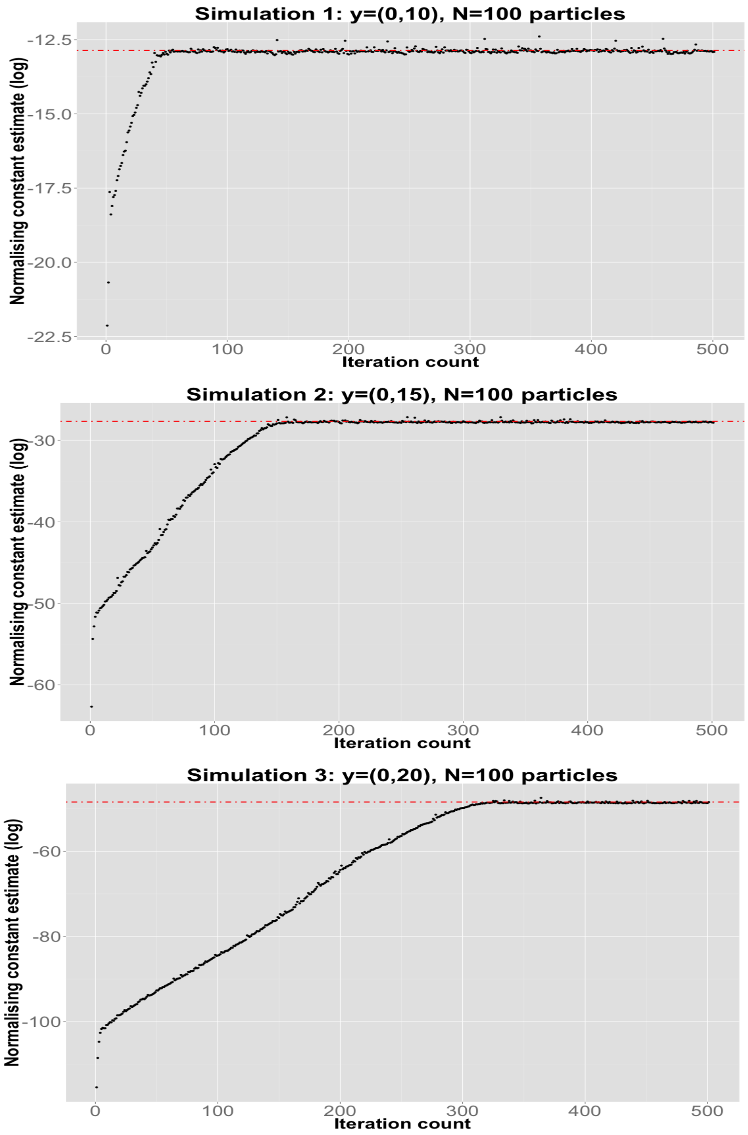
Notice that especially with the first two iterations the average resampling count drops significantly: this is important because a low resampling rate, other than reducing the additional noise introduced at each selection iteration, mitigates the degeneracy of the smoothing weights therefore allowing us to compute fixed-lag smoothing distributions accurately, for example.

8.1.2 Extreme Observation Example

In the last section we showed how the iAPF presents some appreciable improvement compared to a BPF with the same number of particles N. If we compare the computational cost though, given a Linear Gaussian model with parameters such as those in the previous example, it can be more efficient to increase the number of particles and run a bootstrap filter than iterating the look-ahead procedure with fewer particles. Nonetheless we showed that the algorithm effectively obtains good approximations for the look-ahead functions in this setting and we can find very similar results for other HMMs: the use of such a look-ahead scheme could be preferable in cases where future observations are highly informative relative to the latent process evolution.

We consider a Linear Gaussian model in one dimension defined by $\mu(\cdot) = \mathcal{N}(\cdot; 0, 1)$, $f(x, \cdot) = \mathcal{N}(\cdot; x, 1)$ and $g(x, \cdot) = \mathcal{N}(\cdot; x, 0.5)$ and instead of taking a sample from the model we set three very short sequences of observations $y^1_{1:2} = (0, 10)$, $y^2_{1:2} = (0, 15)$ and $y^3_{1:2} = (0, 20)$ so that, because of the big jump between the pair of observations in each sequence, they represent potential extreme realisations of the observation process. In the following plots we show the evolution of the normalising constant estimate through a total of 500 iterations of the algorithm and we use a total of $N = 100$ particles at every iteration. The dotted red line represents the truth estimate, derived via Kalman filtering.

Figure 8.2: Extreme observation filtering simulations: normalising constants estimates


Simulation 1: y=(0,10), N=100 particles

Simulation 2: y=(0,15), N=100 particles

Simulation 3: y=(0,20), N=100 particles

In this setting our algorithm dramatically outperform the bootstrap. With the BPF, billions of particles do not suffice to estimate the likelihood of such an extreme sequence of observations, as the model transition fails to explore the state space efficiently when used as proposal distribution. With our iterative scheme, the subsequent waves of particles assume gradually the behaviour of perfect look-ahead particles as the approximations of the look-ahead proposals become more accurate: the estimate reaches convergence after some iterations.

The execution time is still contained due to the small number of particles in use. If we repeat the experiment using the stopping rule of Algorithm 6.2, the iAPF on average stops after 71, 136 and 336 iterations (standard deviations of the sample equal to 9, 24 and 41) for the instances with $y = (0, 10)$, $y = (0, 15)$ and $y = (0, 20)$ respectively (averages and standard deviations computed over 20 run of the algorithm). In all but a couple of cases for which we have a final iteration with $N = 200$ particles, the starting value of $N_0 = 100$ particles is never increased.

It is true that we are looking at observation process realisations which are arguably too extreme and such a jump will "never" happen simulating from the defined model. We give some motivations to look at such extreme sequences.

- Most of the times the real dynamic of the system is too complex to be captured entirely, nonetheless a model has to be superimposed in order to describe the data. Under this problem known as model misspecification, a non-extreme sequence of observations could become extreme under the superimposed model.

- In many cases we are interested in computing the likelihood associated with a sequence of observations given different model structures and parameters, therefore sequences of extreme observations become more likely. A particular case of this is when the HMM is expressed in terms of a (possibly multidimensional) parameter θ. Many Monte Carlo schemes (see for example Andrieu & Roberts (2009), Andrieu et al. (2010)) that aim at estimating θ rely on a accurate approximation for the normalising constants L: a possible way to do this is to run a Metropolis-Hastings algorithm on the parameter state space and in this case normalising constant estimates appear in the acceptance ratio. If the MH algorithm explores exhaustively the parameter state space it will be necessary to compute normalising constant estimates for arbitrarily extreme (given the parameter configuration) sequence of observations.

- Increasing the dimensionality of the latent process and the observation process quickly leads to a drop in the observation likelihood: the use of the BPF can be problematic in a high-dimensional setting where any sequence of observations can be considered extreme. These preliminary results with extreme observations in one dimension suggest that the iAPF could show interesting performance also for non extreme observation sequences in high dimensional settings. We will look at some examples in this setting with the parametric approach in the following section.

8.1.3 Kernel density estimate vs parametric approach

The obvious advantage of the kernel density estimates approach with respect to the parametric approach is that we do not need to pick the look-ahead functions from a predefined class of functions i.e. we do not need to make any preliminary assumption about the shape of the look-ahead functions $(\psi_1, \ldots, \psi_T)$. On the other hand this method is computationally more expensive. Even selecting a subset of $\sqrt{N}$ particles for the backward recursion as described in Section 7.2 the total computational cost of the algorithm is of order $\mathcal{O}\left(N\sqrt{N}\right)$. Furthermore, while kernel density estimation can be performed in any number of dimensions, in practice the curse of dimensionality causes its performance to degrade in high dimensions. The convergence rate of the kernel density estimate approximation scales exponentially with dimension thus the number of particles needed for a high dimensional model could be prohibitive. From the next section on we will focus on the parametric approach as it presents a much higher efficiency in all our applications, that is a much lower empirical variance of the normalising constant estimates achieved with a similar execution time. When the optimal look-ahead functions are unimodal, choosing Ψ to be the set of Gaussian densities parametrised by their mean and variance/covariance matrix is a simple and natural choice. In this unimodal setting we expect the parametric approach relative to the class of Gaussian functions to be more efficient in most situations than the kernel density estimate approach. However, consider for instance the SSM defined for $t \in 1:T$, $T = 10$, by the initial and transition densities

$$\mu(\cdot) = \mathcal{N}(\cdot; 0, 1) \quad f(x, \cdot) = \mathcal{N}(\cdot; x, 1)$$

with state space $\mathsf{X} = \mathbb{R}$ and the potential $g(x) = \mathbb{1}_{[-2,-1]}(x) + \mathbb{1}_{[1,2]}(x)$ taking values in $\mathsf{Y} = \{0, 1\}$. For this toy model all the optimal look-ahead functions ψ_t^* for $t \in 1:T-1$ are symmetric and bimodal. If we consider the class of Gaussian densities as the class Ψ of look-ahead functions for the parametric optimisation approach (like we did in all our applications), we will not find a good representative $\boldsymbol{\psi}$ of the optimal sequence of look-ahead functions $\boldsymbol{\psi}^*$, as we are trying to approximate the bimodal function ψ_t^* with a unimodal function ψ_t, for all $t \in 1 : T-1$. The corresponding iAPF retains the unbiasedness property of the normalising constant estimates, but it can easily perform worse than the BPF. In the case where the iAPF presents low efficiency, it is easy to detect whether this is due to a high discrepancy between $\boldsymbol{\psi}$ and $\boldsymbol{\psi^*}$. In particular we can plot the two sets of points $(x_t^i, \psi_t^i)_{i\in 1:N}$ and $(x_t^i, \psi_t\left(x_t^i\right))_{i\in 1:N}$ (as defined in Algorithm 7.3), relative to ψ_t^* and ψ_t respectively,_and compare the two resulting dotted surfaces for any $t \in 1 : T$. A possible approach to deal with an excessive discrepancy between approximated and actual optimal look-ahead functions is to consider a different class of density functions for Ψ. For this toy model, for instance, we could choose Ψ to be the class of mixture of Gaussians with two components. This solution, however, is strictly model dependent. The main advantage of the kernel density estimate approach is that no tailoring of the class Ψ is required, which can be necessary for the parametric approach in general scenarios and especially in the case where the functions in the optimal look-ahead sequence $\boldsymbol{\psi^*}$ are multimodal.

8.2 Parametric approach

The parametric approach of Section 7.3 depending on the chosen class of look-ahead function Ψ_θ and minimisation routine can have a computational cost of order $\mathcal{O}\left(N\right)$. It can also be robust for high dimensional model provided that the class of functions Ψ_θ contains elements close enough to the perfect look-ahead functions in the sequence $\boldsymbol{\psi}^*$. In all our applications we choose Ψ_θ to be the set of all Gaussian density functions parametrised by their mean (mean vector) and variance (variance/covariance matrix). This choice led to good results partly because in all our applications we expect the perfect look-ahead functions to be unimodal. Clearly the choice of Ψ_θ is not limited to Gaussian densities and for different applications it could be worth it to design Ψ_θ more carefully, for example as the class of Gaussian mixtures with m components, with $m > 1$. In all the following simulations the stopping rule of Algorithm 6.2 applies.

8.2.1 Relative variance of approximations of Z when d is large

We consider a family of Linear Gaussian models where $m = \mathbf{0}$, $\Sigma = B = C = D = I_d$ and $A_{ij} = \alpha^{|i-j|+1}$, $i, j \in \{1, \ldots, d\}$ for some $\alpha \in (0, 1)$. Our first comparison is between the relative errors of the approximations $\hat{Z}$ of $L = Z$ using the iAPF, the BPF and the FA-APF. We consider configurations with $d \in \{5, 10, 20, 40, 80\}$ and $\alpha = 0.42$ and we simulated a sequence of $T = 100$ observations $y_{1:T}$ for each configuration. We ran 1000 replicates of the three algorithms for each configuration and report box plots of the ratio $\hat{Z}/Z$ in Figure 8.3.

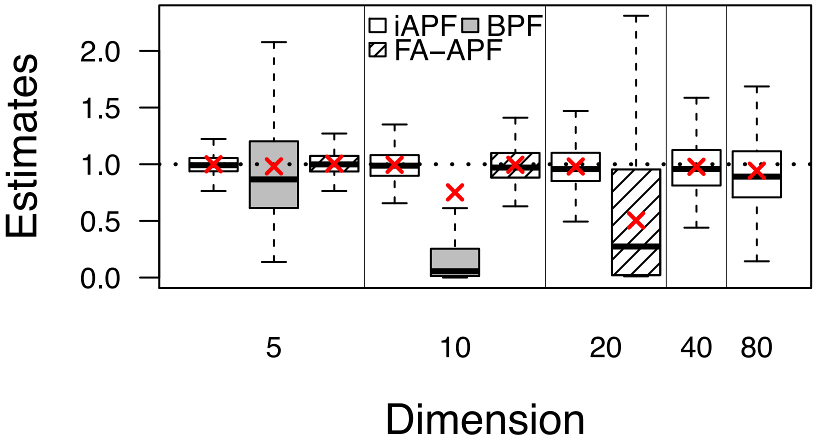


Figure 8.3: Box plots of $\hat{Z}/Z$ for different dimensions using 1000 replicates. The crosses indicate the mean of each sample.

For all the simulations we ran an iAPF with $N_0 = 1000$ starting particles, a BPF with $N = 10000$ particles and an FA-APF with $N = 5000$ particles. In each dimension the BPF and FA-APF both had slightly larger average computational times than the iAPF with these configurations. The average number of particles for the final iteration of the iAPF was greater than N_0 only in dimensions $d = 40$ (1033) and $d = 80$ (1142). For $d > 10$, it was not possible to obtain reasonable estimates with the BPF in a feasible computational time (similarly for the FA-APF for $d > 20$). The standard deviation of the samples and the average resampling count across the chosen set of dimensions are reported in Tables 8.3–8.4.

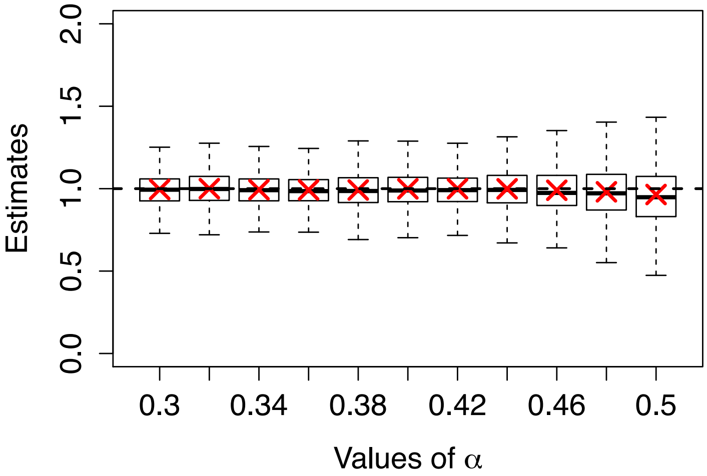
Table 8.3: Empirical standard deviation of the quantity $\hat{Z}/Z$ using 1000 replicates

Dimension	5	10	20	40	80
iAPF	0.09	0.14	0.19	0.23	0.35
BPF	0.51	6.4	-	-	-
FA-APF	0.10	0.17	0.53	-	-

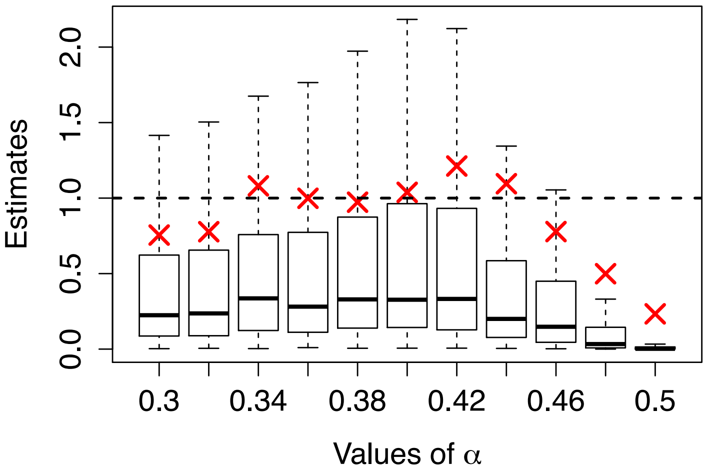
Table 8.4: Average resampling count for the 1000 replicates

Dimension	5	10	20	40	80
iAPF	6.93	15.11	27.61	42.41	71.88
BPF	99	99	-	-	-
FA-APF	26.04	52.71	84.98	-	-

Fixing the dimension $d = 10$ and the simulated sequence of observations $y_{1:T}$ with $\alpha = 0.42$, we now consider the variability of the relative error of the estimates of the marginal likelihood of the observations using the iAPF and the BPF for different values of the parameter $\alpha \in \{0.3, 0.32, \ldots, 0.48, 0.5\}$. In Figure 8.4, we report box plots of $\hat{Z}/Z$ in 1000 replications. For the iAPF, the length of the boxes are significantly less variable across the range of values of α. In this case, we used $N = 50000$ particles for the BPF, giving a computational time at least five times larger than that of the iAPF. This demonstrates that the approximations of the marginal likelihood $L(\alpha)$ provided by the iAPF are relatively insensitive to small changes in α, in contrast to the BPF. Similar simulations, which we do not report, show that the FA-APF for this problem performs slightly worse than the iAPF at double the computational time.



(a) iAPF



(b) BPF

Figure 8.4: Box plots of $\hat{Z}/Z$ for different values of the parameter α using 1000 replicates. The crosses indicate the mean of each sample.

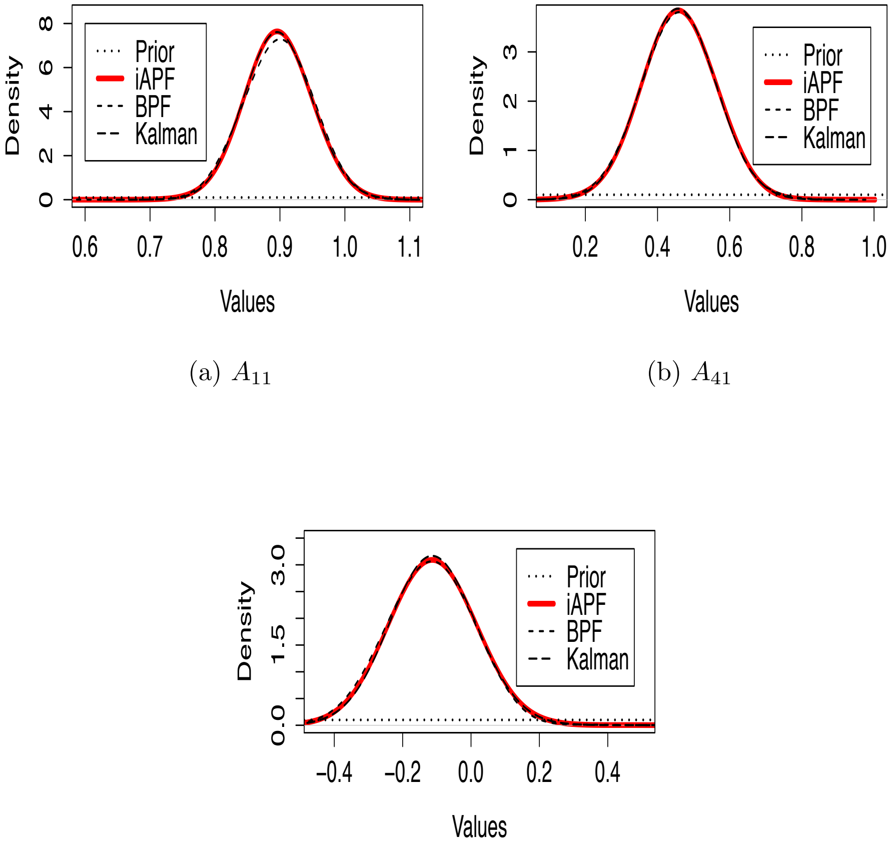
8.2.2 Particle marginal Metropolis–Hastings

We consider a Linear Gaussian model with $m = \mathbf{0}$, $\Sigma = B = C = I_d$, and $D = \delta I_d$ with $\delta = 0.25$. We used the lower-triangular matrix

$$A = \begin{pmatrix} 0.9 & 0 & 0 & 0 & 0 \\ 0.3 & 0.7 & 0 & 0 & 0 \\ 0.1 & 0.2 & 0.6 & 0 & 0 \\ 0.4 & 0.1 & 0.1 & 0.3 & 0 \\ 0.1 & 0.2 & 0.5 & 0.2 & 0 \end{pmatrix},$$

and simulated a sequence of $T = 100$ observations. Assuming only that A is lower triangular, for identifiability, we performed Bayesian inference for the 15 unknown parameters $\{A_{i,j} : i, j \in \{1, \ldots, 5\}, j \leq i\}$, assigning each parameter an independent uniform prior on $[-5, 5]$. From the initial point $A_1 = I_5$ we ran three Markov chains $A_{1:L}^{\text{BPF}}$, $A_{1:L}^{\text{iAPF}}$ and $A_{1:L}^{\text{Kalman}}$ of length $L = 300000$ to explore the parameter space, updating one of the 15 parameters components at a time with a Gaussian random walk proposal with variance 0.1. The chains differ in how the acceptance probabilities are computed, and correspond to using unbiased estimates of the marginal likelihood obtained from the BPF, iAPF or the Kalman filter, respectively. In the latter case, this corresponds to running a Metropolis–Hastings (MH) chain by computing the marginal likelihood exactly. We started every run of the iAPF with $N_0 = 500$ particles. The resulting average number of particles used to compute the final estimate was 500.2. The number of particles $N = 20000$ for the BPF was set to have a greater computational time, in this case $A_{1:L}^{\text{BPF}}$ took 50% more time than $A_{1:L}^{\text{iAPF}}$ to simulate. Note that here and in the following examples in the thesis, N_0 is chosen so that there is rarely the need for doubling the number of particles (from preliminary simulations). This represents our guess of something close to the optimal choice of N_0 (since if we choose a larger N_0 the CPU time will increase and if we choose a smaller N_0 then there will need to be doubling, and if this happens many times then it will result in an higher computational time too).

In Figure 8.5, we plot posterior density estimates obtained from the three chains for 3 of the 15 entries of the transition matrix A. The posterior means associated with the entries of the matrix A were fairly close to A itself, the largest discrepancy being around 0.2, and the posterior standard deviations were all around 0.1. A comparison of estimated Markov chain autocorrelations for these same parameters is reported in Figure 8.6, which indicates little difference between the iAPF-PMMH and Kalman-MH Markov chains, and substantially worse performance for the BPF-PMMH Markov chain. The integrated autocorrelation time of the Markov chains provides a measure of the asymptotic variance of the individual chains' ergodic averages, and in this regard the iAPF-PMMH and Kalman-MH Markov chains were practically indistinguishable (based on the empirical integrated autocorrelation time), while the BPF-PMMH performed between 3 and 4 times worse, depending on the parameter. The relative improvement of the iAPF over the BPF does seem empirically to depend on the value of δ. In experiments with larger δ, the improvement was still present but less pronounced than for $\delta = 0.25$. We note that in this example, $\boldsymbol{\psi}^*$ is outside the class of possible $\boldsymbol{\psi}$ sequences that can be obtained using the iAPF: the approximations in $\boldsymbol{\Psi}$ are functions that are constants plus a multivariate normal density with a diagonal covariance matrix whilst the functions in $\boldsymbol{\psi}^*$ are multivariate normal densities whose covariance matrices have non-zero, off-diagonal entries.

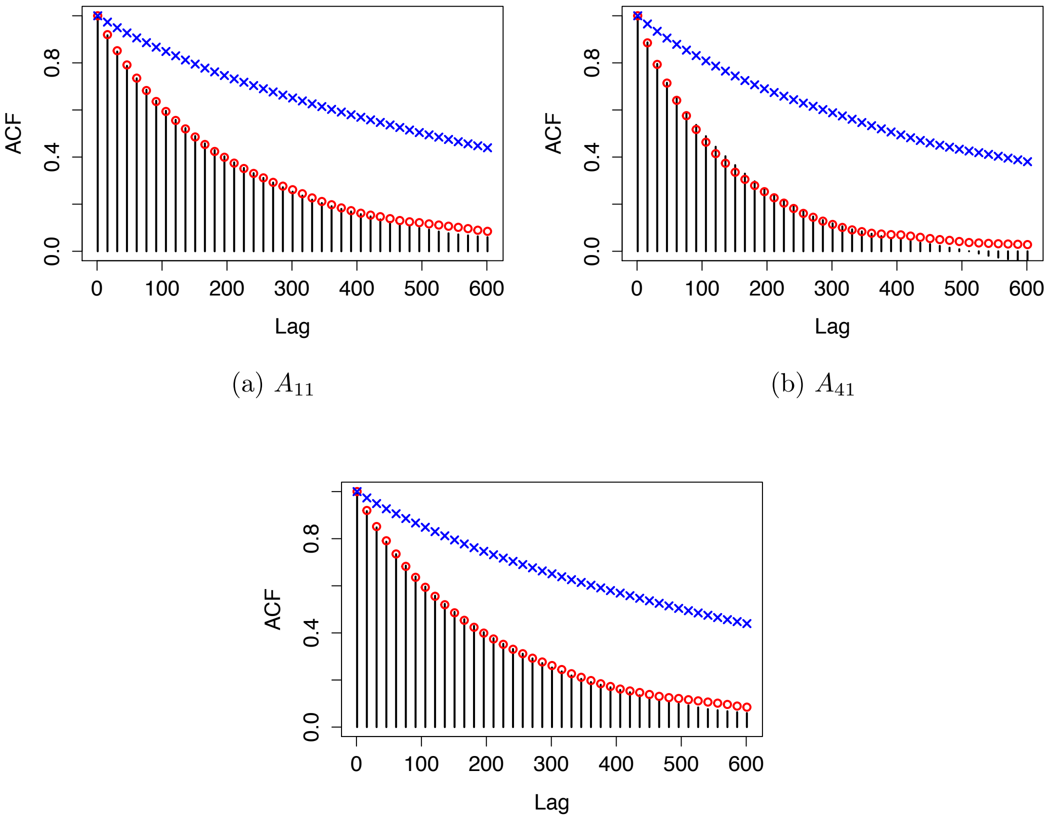


(a) A_{11}

(b) A_{41}

(c) A_{55}

Figure 8.5: Linear Gaussian model: density estimates for the specified parameters from the three Markov chains.



(a) A_{11}

(b) A_{41}

(c) A_{55}

Figure 8.6: Linear Gaussian model: autocorrelation function estimates for the BPF-PMMH (crosses), iAPF-PMMH (solid lines) and Kalman-MH (circles) Markov chains.

In Table 8.5 we also provide the adjusted sample size of the 3 Markov chains associated with each of the 15 parameters, obtained by dividing the length of the chain by the estimated integrated autocorrelation time associated with each parameter.

Table 8.5: Sample size adjusted for autocorrelation for the three methods and the different parameters

	$A_{1,1}$	$A_{2,1}$	$A_{3,1}$	$A_{4,1}$	$A_{5,1}$	$A_{2,2}$	$A_{3,2}$	$A_{4,2}$	$A_{5,2}$	$A_{3,3}$	$A_{4,3}$	$A_{5,3}$	$A_{4,4}$	$A_{5,4}$	$A_{5,5}$
Kalman	2689	2997	603	504	601	3025	2909	765	709	714	513	460	729	640	929
iAPF	2660	2806	574	506	525	3068	2910	755	722	619	501	456	715	652	932
Bootstrap	64	59	18	13	18	72	95	21	27	16	12	16	13	17	27

Chapter 9

Stochastic Volatility Models

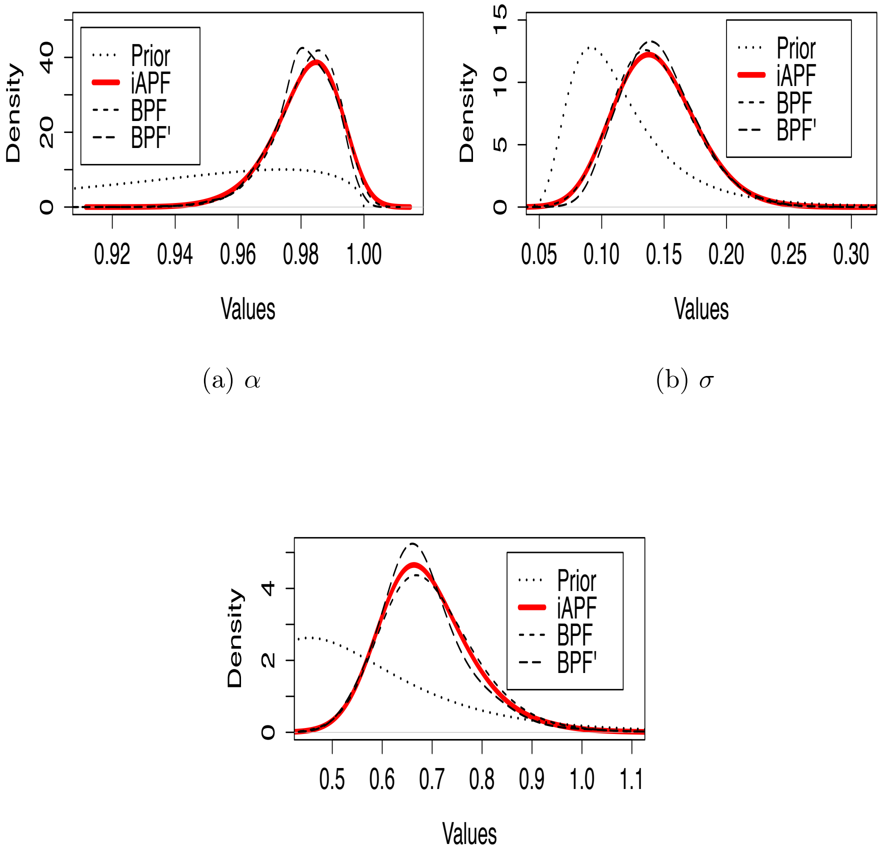
9.1 Univariate stochastic volatility model

A simple stochastic volatility model is defined by $\mu(\cdot) = \mathcal{N}(\cdot; 0, \sigma^2/(1-\alpha)^2)$, $f(x, \cdot) = \mathcal{N}(\cdot; \alpha x, \sigma^2)$ and $g(x, \cdot) = \mathcal{N}(\cdot; 0, \beta^2 \exp(x))$, where $\alpha \in (0,1)$, $\beta > 0$ and $\sigma^2 > 0$ are statistical parameters (see, e.g., Kim et al. 1998). To compare the efficiency of the iAPF and the BPF within a PMMH algorithm, we analyzed a sequence of $T = 945$ observations $y_{1:T}$, which are mean-corrected daily returns computed as

$$y_t = 100 \left[(\log r_{t+1} - \log r_t) - \frac{1}{T} \sum_{i=1}^{T} (\log r_{i+1} - \log r_i) \right]$$

from the weekday close exchange rates $r_{1:T+1}$ for the pound/dollar from 1/10/81 to 28/6/85. This data has been previously analyzed using different approaches, e.g. in Harvey et al. (1994) and Kim et al. (1998).

We wish to infer the model parameters $\theta = (\alpha, \sigma, \beta)$ using a PMMH algorithm and compare the two cases where the marginal likelihood estimates are obtained using the iAPF and the BPF. Following Kim et al. (1998), we placed independent inverse Gamma prior distributions $\mathcal{IG}\,(2.5, 0.025)$ and $\mathcal{IG}\,(3, 1)$ on σ^2 and β^2, respectively, and an independent Beta$\,(20, 1.5)$ prior distribution on the transition coefficient α. Also based on Kim et al. (1998) we used $(\alpha_0, \sigma_0, \beta_0) = \left(0.95, \sqrt{0.02}, 0.5\right)$ as the starting point of the three chains: $X_{1:L}^{\text{iAPF}}$, $X_{1:L}^{\text{BPF}}$ and $X_{1:L'}^{\text{BPF}'}$ where L and L' are the lengths of the first two chains and the third chain respecitvely. All the chains updated one component at a time with a Gaussian random walk proposal with variances $(0.02, 0.05, 0.1)$ for the parameters (α, σ, β). $X_{1:L}^{\text{iAPF}}$ has a total length of $L = 150000$ and for the estimates of the marginal likelihood that appear in the acceptance probability we use the iAPF with $N_0 = 100$ starting particles. For $X_{1:L}^{\text{BPF}}$ and $X_{1:L'}^{\text{BPF}'}$ we use BPFs: $X_{1:L}^{\text{BPF}'}$ is a shorter chain with more particles ($L = 150000$ and $N = 1000$) while $X_{1:L'}^{\text{BPF}'}$ is a longer chain with fewer particles ($L = 1500000$, $N = 100$). All chains required similar running time overall to simulate. Figure 9.1 shows very similar estimated marginal posterior densities for the three parameters using the different chains.



(a) α

(b) σ

(c) β

Figure 9.1: Stochastic Volatility model: PMMH density estimates for each parameter from the three chains.

In Table 9.1 we provide the adjusted sample size of the Markov chains associated with each of the parameters. We can see an improvement using the iAPF, although we note that the BPF-PMMH algorithm appears to be fairly robust to the variability of the marginal likelihood estimates in this particular application.

Table 9.1: Sample size adjusted for autocorrelation for each parameter from the three chains.

	α	σ^2	β
iAPF	3620	3952	3830
BPF	2460	2260	3271
BPF'	2470	2545	2871

Since particle filters provide approximations of the marginal likelihood in HMMs, the iAPF can also be used in alternative parameter estimation procedures, such as simulated maximum likelihood (Lerman & Manski 1981, Diggle & Gratton 1984). The use of particle filters for approximate maximum likelihood estimation (see, e.g., Kitagawa 1998, Hürzeler & Künsch 2001) has recently been used to fit macroeconomic models (Fernández-Villaverde & Rubio-Ramírez 2007). In Figure 9.2 we show the variability of the BPF and iAPF estimates of the marginal likelihood at points in a neighborhood of the approximate MLE of $(\alpha, \sigma, \beta) = (0.984, 0.145, 0.69)$. The iAPF with $N_0 = 100$ particles used 100 particles in the final iteration to compute the likelihood in all simulations, and took slightly more time than the BPF with $N = 1000$ particles, but far less time than the BPF with $N = 10000$ particles. The results indicate that the iAPF estimates are significantly less variable than their BPF counterparts, and may therefore be more suitable in simulated maximum likelihood approximations.

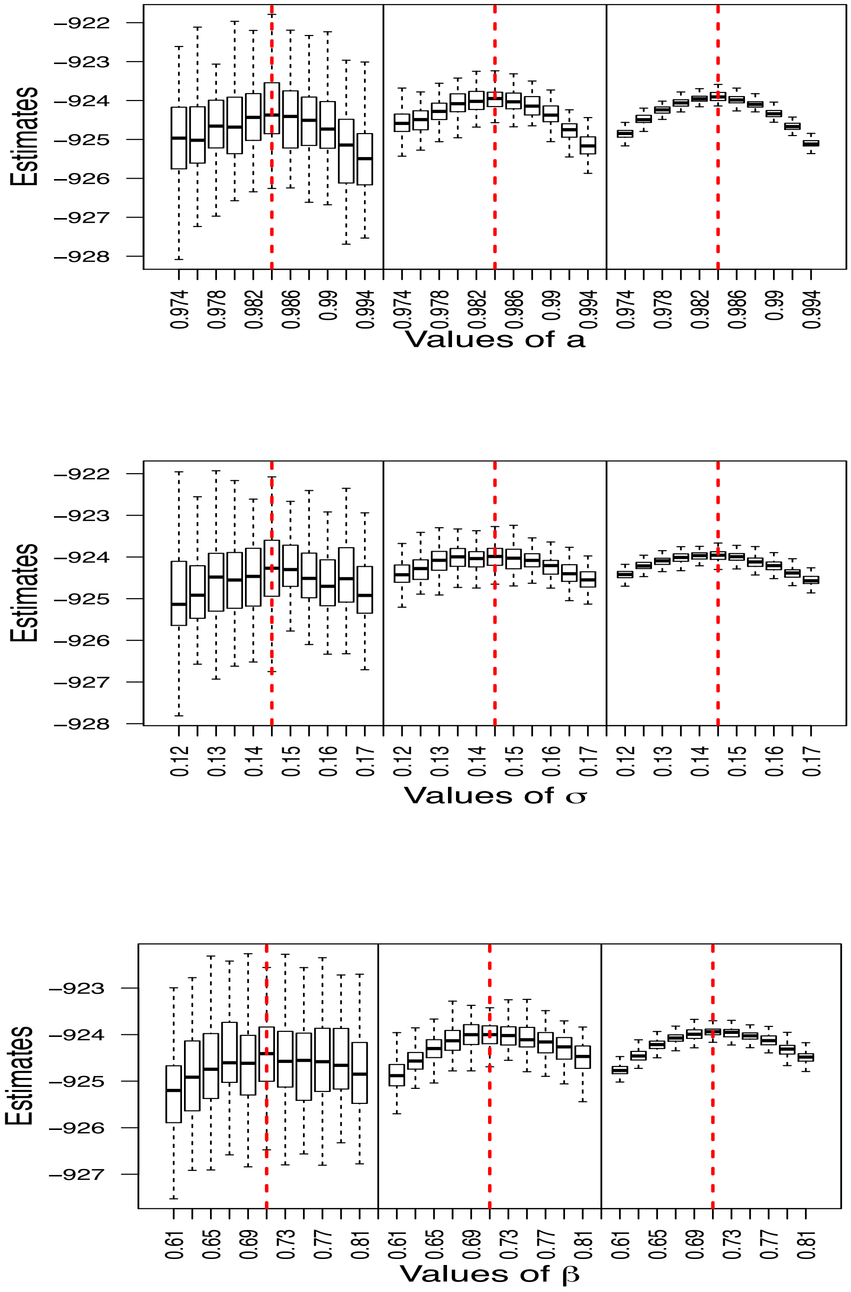


Figure 9.2: log-likelihood estimates in a neighborhood of the MLE. Boxplots correspond to 100 estimates at each parameter value given by three particle filters, from left to right: BPF ($N = 1000$), BPF ($N = 10000$), iAPF ($N_0 = 100$).

9.2 Multivariate stochastic volatility model

We consider a version of the multivariate stochastic volatility model defined for $\mathsf{X} = \mathbb{R}^d$ by $\mu(\cdot) = \mathcal{N}(\cdot; m, U_\star)$, $f(x, \cdot) = \mathcal{N}(\cdot; m + \mathrm{diag}(\phi)\,(x - m)\,, U)$ and $g(x, \cdot) = \mathcal{N}\left(\cdot; 0, \exp\left(\mathrm{diag}\left(x\right)\right)\right)$, where $m, \phi \in \mathbb{R}^d$ and the covariance matrix $U \in \mathbb{R}^{d \times d}$ are statistical parameters. The matrix $U_\star$ is the stationary covariance matrix associated with (ϕ, U). This is the *basic MSV model* in Chib et al. (2009, Section 2), with the exception that we consider a non diagonal transition covariance matrix U and a diagonal observation matrix.

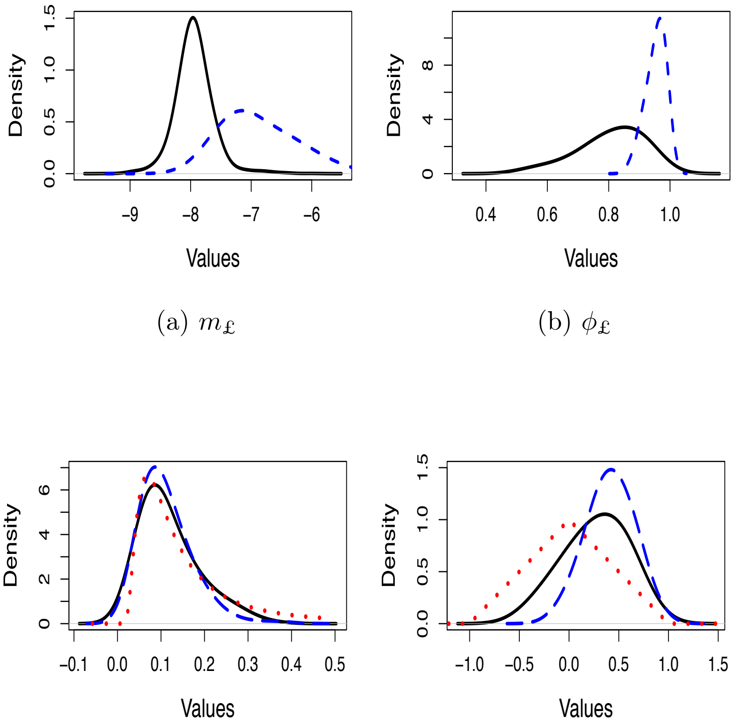
We analyzed two 20-dimensional sequences of observations $y_{1:T}$ and $y'_{1:T'}$, where $T = 102$ and $T' = 90$. The sequences correspond to the monthly returns for the exchange rate with respect to the US dollar of a range of 20 different international currencies, in the periods 3/2000–8/2008 ($y_{1:T}$, pre-crisis) and 9/2008–2/2016 ($y'_{1:T'}$, post-crisis), as reported by the Federal Reserve System (available at `http://www.federalreserve.gov/releases/h10/hist/`). We infer the model parameters $\theta = (m, \phi, U)$ using the iAPF to obtain marginal likelihood estimates within a PMMH algorithm. A similar study using a different approach and with a set of 6 currencies can be found in Liu & West (2001).

The aim of this study is to showcase the potential of the iAPF in a scenario where, due to the relatively high dimensionality of the state space, the BPF systematically fails to provide reasonable marginal likelihood estimates in a feasible computational time. To reduce the dimensionality of the parameter space we consider a band diagonal covariance matrix U with non-zero entries on the main, upper and lower diagonals. We placed independent inverse Gamma prior distributions with mean 0.2 and unit variance on each entry of the diagonal of U, and independent symmetric triangular prior distributions on $[-1, 1]$ on the correlation coefficients $\rho \in \mathbb{R}^{19}$ corresponding to the upper and lower diagonal entries. We place independent $\mathrm{Uniform}(0, 1)$ prior distributions on each component of ϕ and an improper, constant prior density for m. This results in a 79-dimensional parameter space. As the starting point of the chains we used $\phi_0 = 0.95 \cdot \mathbf{1}$, $\mathrm{diag}(U_0) = 0.2 \cdot \mathbf{1}$ and for the 19 correlation coefficients we set $\rho_0 = 0.25 \cdot \mathbf{1}$, where $\mathbf{1}$ denotes a vector of 1s whose length can be determined by context. Each entry of m_0 corresponds to the logarithm of the standard deviation of the observation sequence of the relative currency.

We ran two Markov chains $X_{1:L}$ and $X'_{1:L}$, corresponding to the data sequences $y_{1:T}$ and $y'_{1:T'}$, both of them updated one component at a time and each with a Gaussian random walk proposal with standard deviations equal to $(0.2 \cdot \mathbf{1}, 0.005 \cdot \mathbf{1}, 0.02 \cdot \mathbf{1}, 0.02 \cdot \mathbf{1})$ for the parameters $(m, \phi, \operatorname{diag}(U), \rho)$. The total number of updates for each parameter is $L = 12000$ and the iAPF with $N_0 = 500$ starting particles is used to estimate marginal likelihoods within the PMMH algorithm. In Figure 9.3 we report the estimated smoothed posterior densities corresponding to the parameters for the Pound Sterling/US Dollar exchange rate series. Most of the posterior densities are different from their respective prior densities, and we also observe qualitative differences between the pre and post crisis regimes. For the same parameters, sample sizes adjusted for autocorrelation are reported in Table 9.2. Considering the high dimensional state and parameter spaces, these are satisfactory. In the later steps of the PMMH chain, we recorded an average number of iterations for the iAPF of around 5 and an average number of particles in the final $\boldsymbol{\psi}$-APF of around 502.

Table 9.2: Sample size adjusted for autocorrelation.

	$m_{£}$	$\phi_{£}$	$U_{£}$	$U_{£,€}$
pre-crisis	408	112	218	116
post-crisis	175	129	197	120



(a) $m_{£}$

(b) $\phi_{£}$

(c) $U_{£}$

(d) $U_{£,€}$

Figure 9.3: Multivariate stochastic volatility model: density estimates for the parameters related to the Pound Sterling. Pre-crisis chain (solid line), post-crisis chain (dashed line) and prior density (dotted line). The prior densities for (a) and (b) are constant.

The aforementioned qualitative change of regime seems to be evident looking at the difference between the posterior expectations of the parameter m for the post-crisis and the pre-crisis chain, reported in Figure 9.4. The parameter m can be interpreted as the period average of the mean-reverting latent process of the log-volatilities for the exchange rate series. Positive values of the differences for close to all of the currencies suggest a generally higher volatility during the post-crisis period.

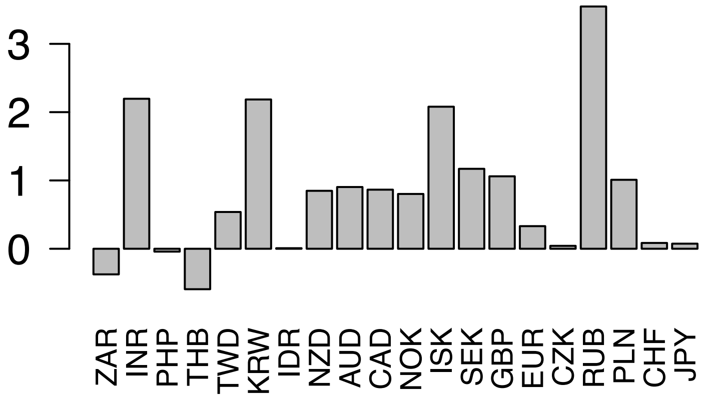


Figure 9.4: Multivariate stochastic volatility model: differences between post-crisis and pre-crisis posterior expectation of the parameter m for the 20 currencies.

Part IV

Examples and Applications to Diffusion Processes

Chapter 10

Diffusion Processes

Diffusion processes are continuous-time Markov processes with almost surely continuous sample paths (see for example Ikeda & Watanabe (1981)): we restrict our attention to the wide class of diffusion processes which can be defined as solutions to stochastic differential equations. Diffusions are a very convenient tool for modeling continuous time data from a wide range of areas such as biology (e.g. Golightly & Wilkinson (2011)), finance (e.g. Ait et al. (2007), Black & Scholes (1973), Cox et al. (1985)), engineering (e.g. Coffey et al. (2004)) bioinformatics (e.g. McAdams et al. (1999)). A diffusion's dynamic is completely characterized by a drift function and a volatility function: much effort has been devoted to developing ways to efficiently estimate the parameters governing these functions from partial and discrete observations.

Consider a Borel drift function $a : \mathbb{R}_+ \times \mathbb{R}^d \longrightarrow \mathbb{R}^d$, a Borel volatility function $b : \mathbb{R}_+ \times \mathbb{R}^d \longrightarrow \mathbb{R}^{d \times d}$ and a stochastic differential equation of the type

$$d\overline{X}_s = a\left(s, \overline{X}_s\right) ds + b\left(s, \overline{X}_s\right) dW_s \tag{10.1}$$

for $s \in [0, S]$, the condition $\overline{X}_0 = \overline{x}_0$ and the multidimensional Wiener process W_s (Revuz & Yor (1994)).

Let $b_k : \mathbb{R}_+ \times \mathbb{R}^d \longrightarrow \mathbb{R}^d$ represent the k-th column of b that is $b_k\left(t, x\right) := \left[b(t, x)\right]_{1:d,k}$. We assume that there exists a K such that

$$\left|a(t, x)\right|^2 + \sum_{k=1}^{d} \left|b_k(t, x)\right|^2 \leq K^2\left(1 + |x|^2\right) \tag{10.2}$$

and

$$\left|a\left(t,x\right)-a\left(t,y\right)\right|+\sum_{k=1}^{d}\left|b_k\left(t,x\right)-b_k\left(t,y\right)\right|\leq K\left|x-y\right| \tag{10.3}$$

for every $x, y \in \mathbb{R}^d$ and every $t \in [0, S]$. Under this condition (see for example Gikhman & Skorokhod (1969, Chapter 8, Theorem 4)) equation (10.1) has a solution $\left(\bar{X}_s\right)_{s\in[0,S]}$ defined on the state space $\mathsf{X} = \mathbb{R}^d$ that is unique up to stochastic equivalence and continuous with probability 1 and with transition probability that we denote by $\mathsf{P}_{t,s}\left(x, A\right)$. We also assume that the functions a and b are 2 times differentiable with bounded derivatives of all orders up to 2, and the matrix $b\left(t,x\right)b'\left(t,x\right)$ is uniformly non-degenerate for every t so that (see Del Moral et al. (2001)) the corresponding transition density that we denote by $\mathsf{p}_{t,s}\left(x, A\right)$ exists and therefore

$$\mathsf{P}_{t,s}\left(x,A\right)=\int_A \mathsf{p}_{t,s}\left(x,y\right)dy$$

for all $0 \leq t \leq s \leq S$, $x \in \mathbb{R}^d$ and $A \in \mathcal{B}\left(\mathbb{R}^d\right)$. Drift and volatility can depend on a set of parameters $\theta \in \Theta$, we leave this dependence implicit for notational simplicity. We are interested in the following problems.

1. Simulation of diffusion bridges, that are sample paths of $\left(\overline{X}_s\right)_{s\in[0,S]}$ conditioning on the event $\left\{\overline{X}_0 = \overline{x}_0, \overline{X}_S = \overline{x}_S\right\}$.

2. Evaluation of transition densities of the form $\mathsf{p}_{0,S}\left(\bar{x}_0, \bar{x}_S\right)$.

3. Evaluation of expectations of the type $\mathbb{E}\left[\phi\left(\overline{X}_s\right) \mid \bar{X}_0 = \bar{x}_0, \overline{X}_S = \overline{x}_s\right]$ for some $s \in [0, S]$ and some regular function ϕ on X.

10.1 Euler–Maruyama approximation

In many applications we are given a sequence of observations from the diffusion path and we face the parameter estimation problem relative to the possibly multidimensional parameter $\theta \in \Theta$ that governs the drift function and volatility function. Closed-form transition densities are rarely available in non trivial cases, therefore likelihood-based inference can be difficult. Some attempts to address this problem include moment based estimation (Chan et al. (1992)) and simulation based methods, see for example Durham & Gallant (2002), which often involve the discretisation of the continuous time diffusion for example through the so called Euler–Maruyama method.

The Euler–Maruyama method is a simple generalisation to stochastic differential equations of the Euler method, a first-order numerical procedure for solving ordinary differential equations with a given initial value (Ascher & Petzold (1998)). Given the discretisation parameter $h > 0$, the Euler–Maruyama approximation to the true solution of (10.1) is the Markov chain $\left(X_t^h\right)_{t \in 1:T+1}$ recursively defined by $X_1^h = \overline{x}_0$ and

$$X_t^h \sim \mathcal{N}\left(X_{t-1}^h + a\left(t-1, X_{t-1}^h\right) h, b\left(t-1, X_{t-1}^h\right) b'\left(t-1, X_{t-1}^h\right) h\right)$$

for $t \in 2 : T+1$, $T = \frac{S}{h}$ and where A' denotes the matrix transponse of the matrix A. Let $\mathbb{P}_h$ denote the law of the Markov chain $\left(X_t^h\right)_{t \in 1:T+1}$. Under appropriate conditions (see Kloeden & Platen (1992, Chapter 10)) we have that

$$\mathbb{E}\left[\sup_{t \in 1:T+1} \left|\bar{X}_{th} - X_t^h\right|\right] \leq M h^{\gamma}$$

where the constant M and γ do not depend on h.

Most Bayesian and likelihood approaches consist of approximating the transition density p with the corresponding Euler–Maruyama discretisation. However, the time step between subsequent observations is typically too large to be used as a time step with the Euler–Maruyama method, therefore the observed low-frequency data is augmented with the introduction of T latent data points between every pair of observations in order for the approximation to become accurate. When the diffusion process is observed without error at fixed time points, the likelihood of the diffusion parameter θ given the sequence of observations takes the form of a product of transition densities, as the transition density of the diffusion process from a given exact observation on is independent from the past. Therefore in this exactly and discretely observed diffusions setting we can focus on the problem of approximating one single transition density.

10.2 Particle filters for diffusion bridges

In order to address the aforementioned problems, for fixed $h > 0$, $S > 0$ and $\overline{x}_S \in \mathsf{X}$, we construct a state-space model from the Euler–Maruyama approximation of (10.1). The initial distribution is given by $\mu = \delta_{\overline{x}_0}$ and the subsequent transition densities by

$$f_t(x_{t-1}, x_t) = \mathcal{N}\left(x_t; x_{t-1} + a(t-1, x_{t-1})h, b(t-1, x_{t-1})b'(t-1, x_{t-1})h\right).$$

The potentials are given by $g_t(x_t) \equiv 1$ for $t \in 1 : T-1$ and

$$g_T(x_T) = \mathcal{N}\left(\overline{x}_S; x_T + a(T, x_T)h, b(T, x_T)b'(T, x_T)h\right).$$

This corresponds to a HMM where no observations of the latent process are available up to time T, when the perfect observation at time $T+1$ yields a noisy pseudo observation. For this model we have that

$$Z_h = \int_{X^T} \mu(x_1) \prod_{t=2}^{T} f_t(x_{t-1}, x_t) g_T(x_T) dx_{1:T}$$

that is the Euler–Maruyama approximation of the transition density $Z = \mathsf{p}_{0,S}(\bar{x}_0, \bar{x}_S)$. Using a particle filter we can get an unbiased estimate Z_h^N of the quantity Z_h which approaches the quantity of interest Z as the parameter h tends to zero (see for example Del Moral et al. (2001)). This is important as for a fine enough discretisation the Euler–Maruyama approximation becomes a reliable representation of the continuous dynamic, therefore we can be interested in making inference on the discretised model instead. Similarly we can use particle filters to produce approximations to smoothing expectations and weighted sample paths of the process $X_{1:T+1}^h$ conditioning on the event $\left\{X_{T+1}^h = \overline{X}_S\right\}$.

10.3 Motivations

Once we reformulate the problem of estimating diffusion transition densities as an SSM normalising constant estimation problem, a wide variety of SMC techniques becomes available. In this setting a BPF corresponds to an Importance

Sampling scheme where a sample $x_{1:T}$ is generated from the proposal distribution with density $\mu(x_1)\prod_{t=2}^{T} f_t(x_{t-1}, x_t)$ and given a weight proportional to $g_T(x_T)$. Z_h^N is an unbiased estimate of Z_h, the Euler-Maruyama approximation that converges to the transition density $\mathsf{p}_{0,S}(\bar{x}_0, \bar{x}_S)$ as h tends to zero. As h decreases the potential $g_T(x_T) = \mathcal{N}(\overline{x}_S; x_T + a(T, , x_T)\, h, b(T, x_T)\, b'(T, x_T)\, h)$ becomes extremely peaked around $\overline{x}_S$, leading to a problem of degeneracy of the importance weights. For this reason when the amount of augmentation is large, that is for small values of h, and a fine partition of the interval $[0, S]$ is required for the Euler-Maruyama approximation to approach the true density, naive SMC schemes can break down. Another critical case is when the density point being estimated is relatively far from the mode of the diffusion process transition density. Note that we are dealing with a degenerate SSM where g_T is the only non constant potential function.

If we set a small enough h, depending on the application, the diffusion nonlinear dynamic can be approximated very well by the Euler-Maruyama Markov process with Gaussian transitions. As the main prerequisite of having Normal transition densities is met, we can think of adapting the iAPF for this problem to obtain a normalising constant estimator more robust with respect to high degrees of augmentation. Once twisted, the SSM model we are considering is not degenerate anymore: the twisted potentials are not constant in general, therefore we can introduce resampling steps within the twisted dynamic. This way we can mitigate the degeneracy problem and improve the accuracy of the estimates of Z_h.

In particular, consider the Euler-Maruyama approximation of the diffusion bridge that follows the SDE (10.1) for $s \in [0, S]$, with conditions $\overline{X}_0 = \overline{x}_0$, $\overline{X}_S = \overline{x}_s$. For the corresponding SSM and the optimal choice of look-ahead functions sequence we have that

$$\psi_T^*(x_T) = g_T(x_T) = \mathcal{N}(\overline{x}_S; x_T + a(T, x_T)\, h, b(T, x_T)\, b'(T, x_T)\, h)$$

and even though the perfect look-ahead function ψ_T^* is not a Gaussian, it can be close to normal when the functions $a(t, \cdot) : \mathbb{R}^d \longrightarrow \mathbb{R}^d$ and $b(t, \cdot) : \mathbb{R}^d \longrightarrow \mathbb{R}^{d \times d}$ are nearly constant for every fixed $t \geq 0$. The look-ahead function backward recursion takes in this case the particular form

$$\psi_t^*\left(x_t\right) = \tilde{\psi}_t^*\left(x_t\right) = \int_{\mathsf{X}} \mathcal{N}\left(x_{t+1}; x_t + a\left(t, x_t\right) h, b\left(t, x_t\right) b'\left(t, x_t\right) h\right) \psi_{t+1}^*\left(x_{t+1}\right) dx_{t+1}$$

for $t \in 1 : T-1$, which is a simplified version of the usual backward recursion where all the potentials of the untwisted SSM but the last one are constant.

Chapter 11

The iAPF Implementation for Diffusion Processes

As the SSM derived from the Euler-Maruyama discretisation has all the desired properties for the application of the iAPF, technically we can run the algorithm in its usual form without any modification. Nonetheless in this section we describe two possible modifications to the iAPF that can significantly enhance its performance in terms of efficiency in this specific diffusion setting.

Recall that the iAPF is based on an iterative procedure where we progressively improve the sequence of look-ahead functions $\psi_{1:T}$ through subsequent waves of exploring particles. Though it is possible to initialise the iterative procedure with a starting sequence $\psi^0_{1:T}$ typically we have too little, if any, a priori knowledge to shape a convenient initial twisted model, and in any case it is difficult to find a way to do so which is applicable in general cases. For this reason our standard approach is not to convey any additional information with the first iAPF iteration, which corresponds to setting $\psi_t \equiv 1$ for all $t \in 1 : T$. This choice is always possible in general but as a consequence the initial iAPF iterations, and in particular the very first one, suffer from the same drawbacks of the BPF. For the untwisted model as $h \downarrow 0$ the final potential function $g_T\left(x_T\right) = \mathcal{N}\left(\bar{x}_S; x_T + a\left(T, x_T\right)h, b\left(T, x_T\right)b'\left(T, x_T\right)h\right)$ becomes arbitrarily peaked around $\bar{x}_S$. For the BPF and therefore for the first iteration of the iAPF this can lead to the problem of the degeneracy of the importance weights. Extremely small values of the importance weights can cause numerical problems in the optimiser routine within the iAPF, but the major issue lies behind the backward recursive estimation procedure. With the backward recursion we estimate a new sequence of possibly enhanced look-ahead functions exploiting the filtering support of the previous iAPF iteration. When g_T is peaked and the first wave of particles is propagated according to the untwisted transition, most if not all the particles $x_T^{1:N}$ at time T are likely to be far from the mode of g_T. With the parametric approach of Section 7.3 at time T we set $\psi_T\left(x_T\right) = \mathcal{N}\left(x_T; m_T, \Sigma_T\right) + c\left(N, m_T, \Sigma_T\right)$ where the parameters m_T and Σ_T are obtained by solving numerically the minimisation problem

$$\left(m_T, \Sigma_T, z_T\right) = \operatorname{argmin}_{(m,\Sigma,z)} \sum_{i=1}^{T} \left(g_T\left(\xi_T^i\right) - z\mathcal{N}\left(\xi_T^i; m, \Sigma\right)\right)^2 + l\left(\xi_t^{1:N}, N, m, \Sigma, \lambda\right).$$

Recall that in the simulations of Part III we set $l \equiv 0$ and we relied on an implicit form of regularisation based on the use of a local optimiser. In this case if we neglect the regularisation term $l\left(\xi_t^{1:N}, N, m, \Sigma, \lambda\right)$ we are effectively fitting a Gaussian distribution to a potential function which is not Gaussian, using points far in the tails of the potential function. This can cause some instability in the estimates of the first sequences of look-ahead functions. Due to the very small values of $g_T\left(x_T^i\right)$ for all $i \in 1:N$, the numerical optimiser can potentially return extreme parameter values that can lead to a $\boldsymbol{\psi}$-APF less efficient than desired. According to the defined stopping rule, the iAPF stops only when the empirical variance of k subsequent normalising constant estimates falls below the specified threshold τ, and for this to happen we expect the look-ahead function parameters to have reached a state of equilibrium. Nonetheless the iAPF scheme keeps doubling the number of particles until this equilibrium is approached and this can lead to an higher stopping time if the first many sequences of look-ahead functions are relatively unstable.

In the next subsections we describe two different approaches that exploit the features of this setting in order to try to address or at least mitigate this problem. With the first method we initialise the iAPF to be applied to an Euler–Maruyama discretisation with the desired small value h with a sequence $\boldsymbol{\psi}^0$ obtained by running a complementary iAPF for the same diffusion bridge but with a much coarse Euler–Maruyama discretisation for which the concentration of g_T is not critical. With the second method we provide a simple specification for the regularisation term to be added to the objective function of the optimisation routine that penalises an excessive discrepancy between ψ_{t_1} and ψ_{t_2} when t_1 is close to t_2. Both methods work for the general class of diffusion as in (10.1) and are not mutually exclusive, we can choose to use either or both depending on the application.

11.1 Complementary iAPF initialisation

Consider the SSM given by the Euler–Maruyama approximation of (10.1) with final condition $\bar{X}_S = \bar{x}_S$ and discretisation step h. For a general SSM it is difficult to come up with a suitable initialising sequence of non-constant look-ahead functions. In the case of diffusions though, there exists a natural SSM with a similar structure to the one considered and that potentially does not present the same difficulties: the SSM given by a more coarse Euler–Maruyama discretisation applied to the very same diffusion bridge. Running the iAPF for this complementary SSM can help us gathering enough information to define an appropriate initialising sequence $\boldsymbol{\psi}^0$ for the original model.

In particular, we can consider the complementary SSM given by the Euler–Maruyama discretisation with approximation step $h' = \kappa h$ and corresponding final time $T' = \frac{S}{h'} = \frac{T}{\kappa}$. We choose $\kappa \in \mathbb{N}$ so that it divides T, that is if the division of T by κ gives an integer number, and in this case we write $\kappa \mid T$ (and $\kappa \nmid T$ if it does not). Assume that κ is such that the corresponding final untwisted potential $g_{T'}\left(x_{T'}\right) = \mathcal{N}\left(\bar{x}_S; x_{T'} + a\left(T, x_{T'}\right)h'; b\left(T, x_{T'}\right)b'\left(T, x_T\right)h'\right)$ is flat enough not to present the aforementioned problem with the degeneracy of the importance weights: a possible approach to obtain such κ is provided at the end of the section. We run an iAPF with N_0 starting particles for the complementary SSM, initialised with a sequence of constant look-ahead functions. From the output of the algorithm we store, together with the estimate $Z^N_{h'}$ of the normalising constant $Z_{h'}$, the final sequence of look-ahead functions $\boldsymbol{\psi}^L$, where L is the number of iterations the algorithm has required to meet the stopping rule. Note that $\boldsymbol{\psi}^L = \psi^L_{1:T'}$ is a sequence of $T' = \frac{T}{\kappa}$ Gaussian distributions that can be completely described by the sequence of means and covariance matrices $\left(m'_{1:T'}, \Sigma'_{1:T'}\right)$ where $\psi^L_{t'}\left(x\right) = \mathcal{N}\left(x; m'_{t'}, \Sigma'_{t'}\right)$ for every $t' \in 1 : T'$. In order to initialise the iAPF for the original SSM we need a longer sequence

$\boldsymbol{\psi}^0 = \psi^0_{1:T}$ or equivalently a set of parameters $(m_{1:T}, \Sigma_{1:T})$. It is natural for those $t \in 1 : T$ such that $\kappa \mid (t-1)$ to set $m_t = m'_{\frac{t-1}{\kappa}+1}$ and $\Sigma_t = \Sigma'_{\frac{t-1}{\kappa}+1}$ because the discrete time indices t and $\frac{(t-1)}{\kappa} + 1$ correspond to the same continuous time $(t-1)\,h = \frac{(t-1)}{\kappa} h'$ for the original and the complementary SSM respectively (see Figure 11.1). The fact that we are dealing with continuous sample paths and that a and b are continuous suggests that we can expect parameters of look-ahead functions relative to close times to be similar. For this reason for those t such that $\kappa \nmid (t-1)$ we set the parameter value at t as an intermediate interpolated value between the same parameter value at t_1 and t_2, where $t_1, t_2 \in 1 : T$ are the closest times to t such that $t_1 < t < t_2$ and $\kappa \mid (t_1 - 1), (t_2 - 1)$. More precisely given $(m_{1:T'}, \Sigma_{1:T'})$ from the iAPF execution for the complementary SSM we set $m_t = m'_{\frac{t-1}{\kappa}+1}$ for all $t \in 1 : T$ such that $\kappa \mid (t-1)$ and

$$m_t = m'_{\left\lfloor \frac{t-1}{\kappa} \right\rfloor + 1} + \left(m'_{\left\lfloor \frac{t-1}{\kappa} \right\rfloor + 2} - m'_{\left\lfloor \frac{t-1}{\kappa} \right\rfloor + 1} \right) \left[(t-1) \mod \kappa \right] h$$

where $a \mod b$ indicates the remainder of the division of a by b. In a similar way we set $\Sigma_t = \Sigma'_{\frac{t-1}{\kappa}+1}$ for all $t \in 1 : T$ such that $\kappa \mid (t-1)$ and

$$\Sigma_t = \Sigma'_{\left\lfloor \frac{t-1}{\kappa} \right\rfloor + 1} + \left(\Sigma'_{\left\lfloor \frac{t-1}{\kappa} \right\rfloor + 2} - \Sigma'_{\left\lfloor \frac{t-1}{\kappa} \right\rfloor + 1} \right) \left[(t-1) \mod \kappa \right] h$$

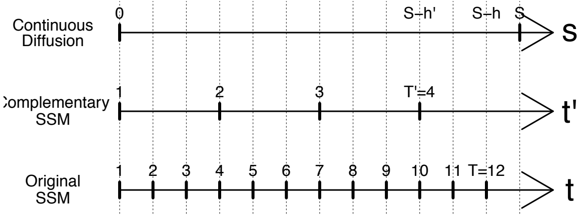
where all the matrix operations are intended entry wise.

Consider for example the diffusion process

$$d\bar{X}_s = \bar{X}_s ds + dW_s$$

with conditions $\bar{X}_0 = 0$ and $\bar{X}_1 = 5$ for $s \in [0, 1]$. Let us say we want to set $h = \frac{1}{12}$, $T = 12$, but we establish this is a too fine discretisation for our problem. We run instead the iAPF corresponding to the discretisation step $h' = \frac{1}{4}$ (therefore $\kappa = 3$) and obtain a sequence of look-ahead functions $\psi_{1:T'}$ where $T' = 4$ corresponding to the continuous time points $(0, 0.25, 0.5, 0.75)$, see Figure 11.1. The parameters of the look-ahead functions in the initialising sequence $\psi^0_{1:T}$ of the original model are obtained through interpolation. In Figure 11.2 we show in black the two look-ahead functions ψ_3 and ψ_4 obtained through the complementary iAPF, we set $\psi^0_7 = \psi_3$ and $\psi^0_{10} = \psi_4$. The red

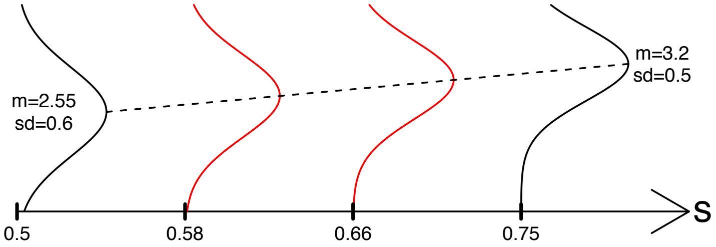
Figure 11.1: Correspondence between diffusion times and Euler-Maruyama discretisations times



look-ahead functions ψ_8^0 and ψ_9^0 in between the continuous time points 0.5 and 0.75 are obtained through interpolation of the mean and standard deviation parameters. Note that this linear scheme is just one possibility and its good performance in our simulations is the reason why more complicated scheme have not been investigated.

We usually have some flexibility on how to set the parameter T that determines the Euler-Maruyama approximation step $h = \frac{S}{T}$. For our simulations we always set $T = 10^{\alpha}$ as an integer power of 10 and for the most challenging cases setting $\kappa = 10$ is always sufficient for providing an initialising sequence $\boldsymbol{\psi}^0$ that can prevent the degeneracy problem. In general though the order of magnitude that T has to have in order for the corresponding Euler–Maruyama discretisation to be a good enough approximation of the continuous diffusion is often unknown. In this case we can very conveniently apply the described procedure in an iterative fashion. We choose an initial very coarse Euler–Maruyama discretisation corresponding to a very small $T = T_1$. We run the iAPF with N starting particles and store the normalising constant estimate $Z^N_{\frac{S}{T_1}}$ of $Z_{\frac{S}{T_1}}$ along with the final sequence of look-ahead functions $\boldsymbol{\psi}^{(T_1)}$. We use the sequence $\boldsymbol{\psi}^{(T_1)}$

Figure 11.2: Example of look-ahead initialisation. Process $d\bar{X}_s = \bar{X}_s ds + dW_s$, $\bar{X}_0 = 0$, $\bar{X}_1 = 5$.



to define an extended interpolated sequence $\psi_{1:T_2}$ with the described procedure, where $T_2 = \kappa T_1$ and run an iAPF with N starting particles and initialised with $\psi_{1:T_2}$ for the SSM corresponding to the Euler-Maruyama discretisation with $T = T_2$. The output $\left(Z^N_{\frac{S}{T_2}}, \boldsymbol{\psi}^{(T_2)}\right)$ of the last iAPF run can be used to replicate the procedure for the discretisation with $T = T_3 = \kappa T_2$ and so on. We can keep refining the discretisation until we are satisfied with the result: a possible stopping rule can be based on the empirical standard deviation of the last j normalising constant estimates.

11.2 Regularisation

Recall that with the parametric optimisation approach, in order to define ψ_t we solve numerically the minimisation problem

$$\text{argmin}_{(m,\Sigma,\lambda)} \sum_{i=1}^{N} \left[\mathcal{N}\left(\xi_t^i; m, \Sigma\right) - \lambda \psi_t^i\right] + l\left(\xi_t^{1:N}, N, m, \Sigma, \lambda\right)$$

where $\xi_t^{1:N}$ is the filtering support at time t at the previous iAPF iteration and the values $\psi_t^{1:N}$ depend on ψ_{t+1} and on the SSM transitions and potentials. For the applications in Part III we neglected the regularisation term $l\left(\xi_t^{1:N}, N, m, \Sigma, \lambda\right)$ as relying on the implicit form of regularisation given by the use of a local optimiser led to good simulation results. In the diffusions setting, when h is small, in the first iAPF iterations the particles $\xi_t^{1:N}$ can be far in the tails of the optimal Gaussian density function that we would like to set as ψ_t. For this reason, even if effort is put in initialising the optimisation routine, the minimiser can fall far from local optima therefore including the regularisation term $l\left(\xi_t^{1:N}, N, m, \Sigma, \lambda\right)$ becomes critical. Recall that we stated in Section7.3 that if we do not include term $l\left(\xi_t^{1:N}, N, m, \Sigma, \lambda\right)$ in the objective function, we have that

$$\lim_{\substack{|m| \to \infty \\ \lambda \to 0}} \sum_{i=1}^{N} \left[\mathcal{N}\left(\xi_t^i; m, \Sigma\right) - \lambda\psi_t^i\right]^2 = 0$$

independently from $\xi_t^{1:N}$ and Σ and therefore an unconstrained optimiser can return a trivial solution. To address this problem we introduce a Tikhonov regularisation term in the objective function. Given that the diffusion paths are continuous we expect the parameters of look-ahead functions relative to close times not to change dramatically. For this reason we add a correction term in the objective function that penalises big Euclidian distances between modes of subsequent look-ahead functions. In particular for the final look-ahead function parameters we set

$$(m_T^*, \Sigma_T^*, \lambda_T^*) = \operatorname{argmin}_{(m,\Sigma,\lambda)} \Sigma_{i=1}^N \left[\mathcal{N}\left(\xi_T^i; m, \Sigma\right) - \lambda\psi_T^i\right]^2 + \frac{c\,(N, m, \Sigma, \lambda)}{h} \left|m - \bar{x}_S\right|$$

and then proceeding backwards recursively

$$(m_t^*, \Sigma_t^*, \lambda_t^*) = \operatorname{argmin}_{(m,\Sigma,\lambda)} \Sigma_{i=1}^N \left[\mathcal{N}\left(\xi_t^i; m, \Sigma\right) - \lambda\psi_t^i\right]^2 + \frac{c\,(N, m, \Sigma, \lambda)}{h} \left|m - m_{t+1}^*\right|$$

where c is a real function. Ideally we would set the term $c\,(N, m, \Sigma, \lambda)$ to grow more slowly than N (i.e. to be $o(N)$). In practice for our simulations we set $c(N, m, \Sigma, \lambda) = N\mathcal{N}\left(m; m, \Sigma\right)$ for simplicity and because at least in our examples this choice is not critical as long as the penalising term scales with $\frac{1}{h}$.

As mentioned before the two methods are not mutually exclusive and one can choose either or both depending on the application. Adding the penalising term to the objective function in the backward recursion has the effect that in general the optimisation will not return the closest fit to the optimal look-ahead function from the chosen class of functions. Nonetheless this second procedure is practically inexpensive and always effective in the examples and applications we consider, therefore this is the one we use in all our simulations if not stated otherwise.

Chapter 12

Examples and Applications

12.1 Exploratory example with a basic diffusion bridge

We perform some simulations related to the basic diffusion bridge defined as the solution of

$$d\bar{X}_s = \alpha ds + \beta dW_s, \tag{12.1}$$

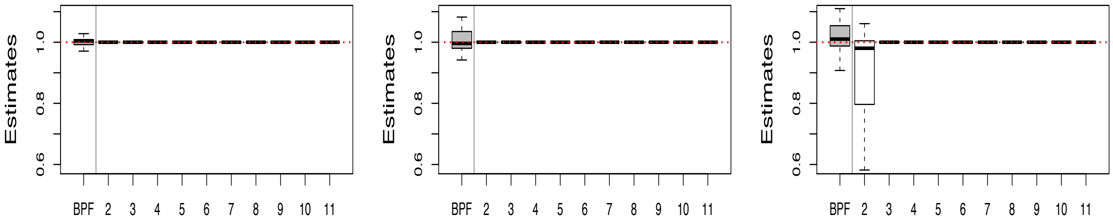
with initial condition $\bar{X}_0 = 0$, final condition $\bar{X}_S = \bar{x}_S$ and the parameters $\alpha \in \mathbb{R}$ and $\beta > 0$. We choose this model as in this case it is straightforward to derive an analytic expression for the transition density that is

$$\mathsf{p}_{t,s}\left(x_t, x_s\right) = \mathcal{N}\left(x_s; x_t + \alpha\left(s-t\right), \beta^2\left(s-t\right)\right)$$

for all $x_t, x_s \in \mathbb{R}$ and any $s > t \geq 0$. The corresponding SSM given by the Euler–Maruyama discretisation with parameter h and $T = \frac{S}{h}$ is defined by $\mu\left(x_1\right) = \delta_0\left(x_1\right)$, $f\left(x_t, x_{t+1}\right) = \mathcal{N}\left(x_{t+1}; x_t + \alpha h; \beta^2 h\right)$ and $g_T\left(x_T\right) = \mathcal{N}\left(\bar{x}_S; x_T + \alpha h, \beta^2 h\right)$. Note that in this case we have that $\psi_T^*\left(x_T\right) = g_T\left(x_T\right) = \mathcal{N}\left(x_T; \bar{x}_S - \alpha h, \beta^2 h\right)$ therefore the perfect final look-ahead function is exactly a normal distribution. We can easily verify that

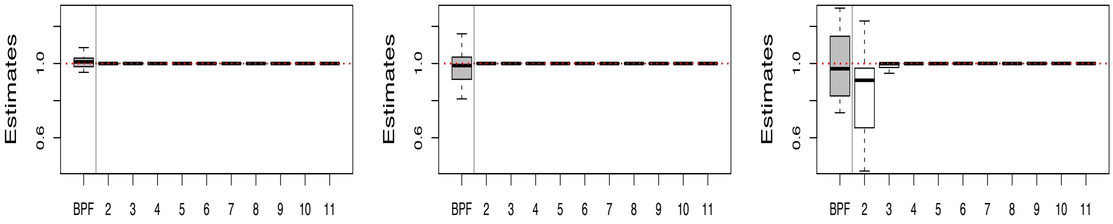
$$\begin{aligned} \psi_{T-1}^*\left(x_{T-1}\right) &= \int_{\mathbb{R}} \mathcal{N}\left(x_T; x_{T-1} + \alpha h, \beta^2 h\right) \mathcal{N}\left(x_T; \bar{x}_S - \alpha h, \beta^2 h\right) dx_T \\ &= \mathcal{N}\left(x_{T-1}; \bar{x}_S - 2\alpha h, 2\beta^2 h\right) \end{aligned}$$

and in general $\psi_t^*\left(x_t\right)=\mathcal{N}\left(x_t;\bar{x}_S-\left(T-t+1\right)\alpha h,\left(T-t+1\right)\beta^2 h\right)$ for all $t\in 1:T-1$ therefore the optimal look-ahead functions are all Gaussians, the class of distributions we use for the parametric optimisation. Note also that equation (12.1) describes a Brownian motion with drift parameter α and scale parameter β with Gaussian transition densities, and therefore the solution of 12.1 is a Brownian bridge once we include the condition $\bar{X}_S=\bar{x}_S$. In this case the Euler–Maruyama discretisation does not introduce any approximation error, it leads to exact solutions for any h and in particular $\mathbb{P}_h\left(g_T\left(X_T^h\right)\mid X_1^h=\bar{x}_0\right)=\mathsf{p}_{0,S}\left(\bar{x}_0,\bar{x}_S\right)$ for any $h>0$. We compare the performances of the BPF and the iAPF in estimating the transition density $Z_h=\mathsf{p}_{0,S}\left(0,0\right)$ for different values of the constant drift parameter α and different values of the step parameter h. In the simulations that follow we fix $\bar{x}_S=0$, $\beta=1$, $S=1$ and we compare the estimates of $\frac{Z_h^N}{Z}$ from a BPF with $N=10000$ particles and from an iAPF with $N_0=200$ particles. The number of particles throughout all of this part is set so that the iAPF computational time is (significantly) lower than that of the BPF. The number of particles is set so that in all the instances the BPF requires higher computational time. For each of the three values of the drift parameter $\alpha\in\{1,2,4\}$ we consider three values of the parameter $h\in\{10^{-1},10^{-2},10^{-3}\}$. The BPF is increasingly challenged as the discretisation becomes finer and for higher values of the drift α, corresponding to more extreme observations (as $\bar{x}_S=0$). With the finest discretisation in (12.3) the BPF systematically fails to provide reasonable estimates. As for the iAPF our stoping rule with $\tau=1$ was met before the sixth iteration in all cases, but we include the boxplots relative to $L=10$ iterations of the algorithm to show that even in this setting the chosen stopping rule seems empirically adequate in detecting the convergence of the iAPF look-ahead functions estimates.



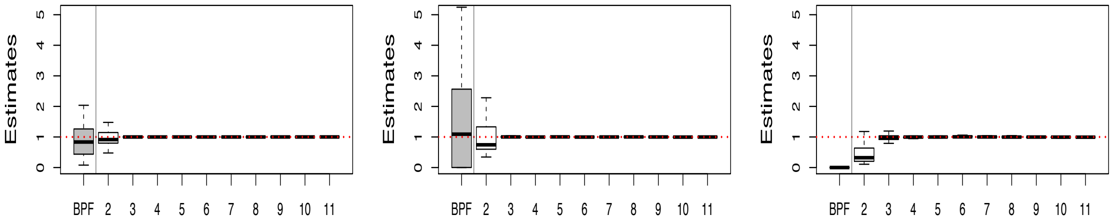
(a) $h = 10^{-1}$ (b) $h = 10^{-2}$ (c) $h = 10^{-3}$

Figure 12.1: Drift parameter $\alpha = 1$. Box plots of the estimates of $\frac{\hat{Z}}{Z}$ for 50 replicates of the BPF and the iAPF.



(a) $h = 10^{-1}$ (b) $h = 10^{-2}$ (c) $h = 10^{-3}$

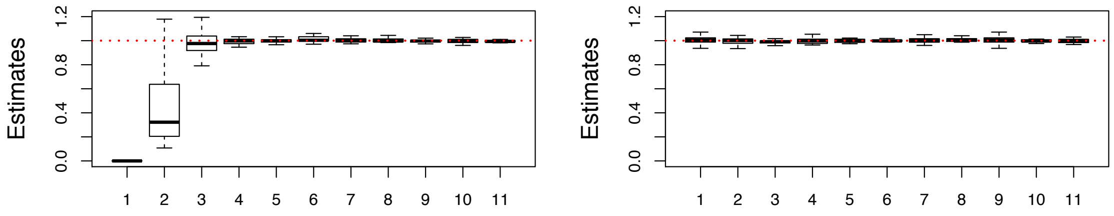
Figure 12.2: Drift parameter $\alpha = 2$. Box plots of the estimates of $\frac{\hat{Z}}{Z}$ for 50 replicates of the BPF and the iAPF.



(a) $h = 10^{-1}$ (b) $h = 10^{-2}$ (c) $h = 10^{-3}$

Figure 12.3: Drift parameter $\alpha = 4$. Box plots of the estimates of $\frac{\hat{Z}}{Z}$ for 50 replicates of the BPF and the iAPF.

If we take a closer look at what happens when $\alpha = 4$ and $h = 10^{-3}$ in Figure 12.4a we can see that it takes at least 3 iterations for the iAPF to provide reasonably accurate estimate. In this case the iAPF still performs well and the computational time of the algorithm is not compromised, but we show how an appropriate initialisation can improve its efficiency. In Figure 12.4b we show the corresponding results for an iAPF where the initial sequence is defined through the interpolation method of the previous section. In Figure 12.5 we show the logarithm of the empirical relative standard deviation for the two algorithms. The execution time of the initialised iAPF is about 50% lower for the 10 iterations of the iAPF, and therefore even lower if we let the iAPF stop according to the usual stopping rule.

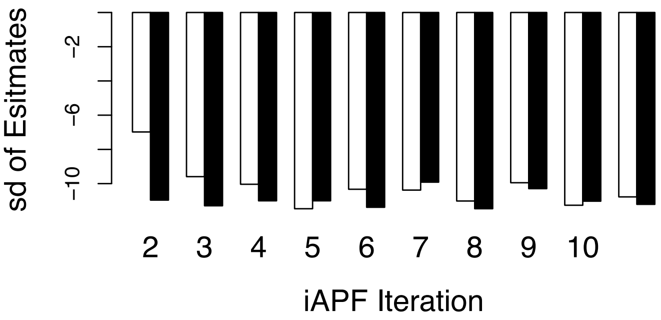


(a) non initialised iAPF

(b) initialised iAPF

Figure 12.4: Box plots of the estimates of $\frac{\hat{Z}}{Z}$ for 50 replicates of the BPF and the iAPF.

Figure 12.5: Logarithm of the relative standard deviations for the non initialised (white) and initialised (black) iAPF



12.2 Sin diffusion

We consider the following diffusion process

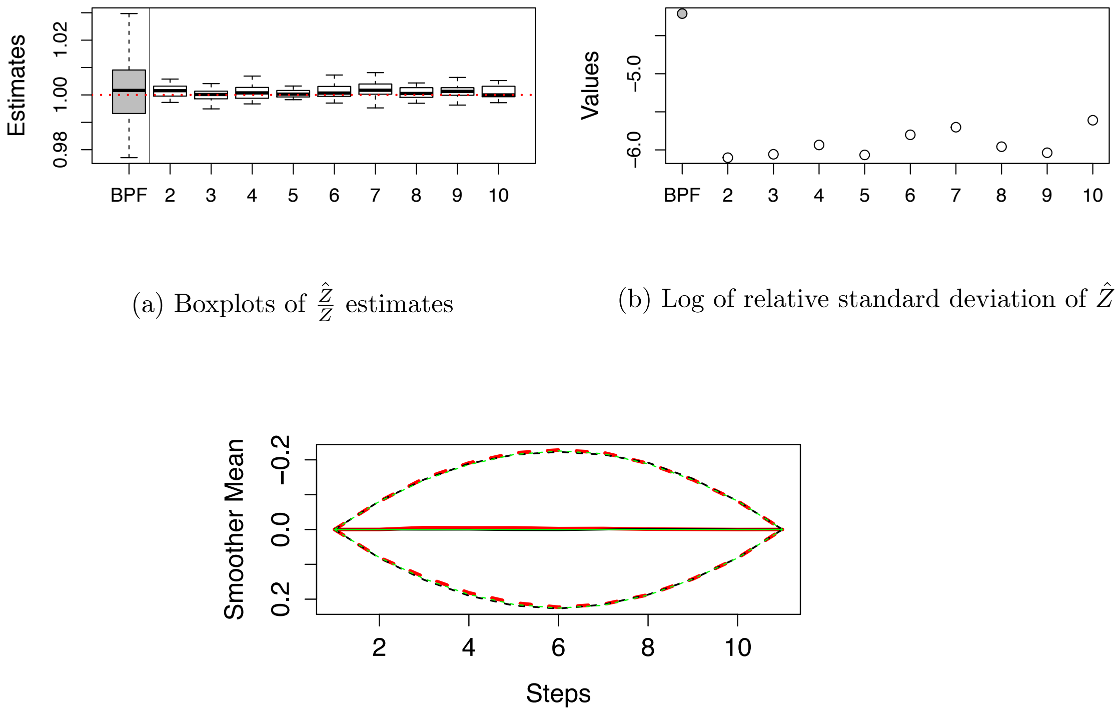
$$d\bar{X}_s = \sin\left(\bar{X}_s\right) ds + dW_s$$

with the conditions $\bar{X}_0 = 0$ and $\bar{X}_1 = c$ on the interval $[0, 1]$. For this diffusion exact simulation is possible and it has been used as an example to showcase the Exact Algorithm in Beskos & Roberts (2005). Note that the estimates from the Exact Algorithm are relative to the continuous diffusion, therefore we can investigate on the impact of discretisation. We run some simulations to compare the efficiency of the iAPF with respect to the BPF and use the Exact Algorithm to check the results. Note that the optimal look-ahead functions $\psi^*_{1:T}$ relative to the SSM given by the Euler–Maruyama discretisation with parameter h are not Gaussian in general, contrary to the previous example with the Brownian bridge. We run simulations for all the combinations of values for the parameters $c \in \left\{0, \pi, \frac{3}{2}\pi\right\}$ and $h \in \{10^{-1}, 10^{-2}, 10^{-3}\}$ and we report in Table 12.1 the standard deviations of $\frac{\hat{Z}}{Z}$ for 50 replicates of the iAPF with $N_0 = 500$ and of the BPF with $N = 10000$. The performance of the BPF becomes worse for finer discretisations, that is for smaller values of h, and for more extreme values of $\bar{x}_S$, that is when $\bar{x}_S$ is far from the mode of the transition density $\mathsf{p}_{0,1}(0, \cdot)$. The iAPF provides better performance than the BPF in all cases, nonetheless the BPF can give reasonably good estimates for $\bar{x}_S = 0$ and, to a lesser extent, for $\bar{x}_S = \pi$. For the case where $\bar{x}_S = \frac{3}{2}\pi$ and for the finer discretisations given by $h = 10^{-2}$ and $h = 10^{-3}$ the BPF fails to provide any sensible estimate, while the iAPF performs very well also for this parameter configurations.

Table 12.1: Logarithm of standard deviation of $\frac{\hat{Z}}{Z}$ for the iAPF and the BPF

	$h = 10^{-1}$	$h = 10^{-2}$	$h = 10^{-3}$
$\bar{x}_S = 0$	iAPF: -5.58 BPF: -4.21	iAPF: -5.70 BPF: -3.42	iAPF: -5.91 BPF: -2.98
$\bar{x}_S = \pi$	iAPF: -4.48 BPF: -3.34	iAPF: -4.14 BPF: -2.38	iAPF: -4.59 BPF: -1.49
$\bar{x}_S = \frac{3}{2}\pi$	iAPF: -3.92 BPF: $+0.45$	iAPF: -4.22 BPF: $-$	iAPF: -3.28 BPF: $-$

We show some additional plots for the three cases corresponding to the parameter pairs $(c, h) = (0, 10^{-1})$ in Figure 12.6, $(c, h) = (\pi, 10^{-2})$ in Figure 12.7 and $(c, h) = \left(\frac{3}{2}\pi, 10^{-3}\right)$ in Figure 12.8 which are expected to represent different degrees of challenge for the BPF as they correspond to increasingly fine discretisations and increasingly extreme realisations. The first two plots in each figure compare the variability of 50 estimates from the BPF and from the iAPF. The third plot represents the estimated smoothing mean for the BPF (black),the iAPF (red) and the Exact Algorithm (green) plus and minus one estimated smoothing standard deviation (dashed lines), that are respectively the expected value and the standard deviation of the process $\left(X_t^h\right)_{t\in 1:T}$ conditioning on $X_{T+1}^h = c$.

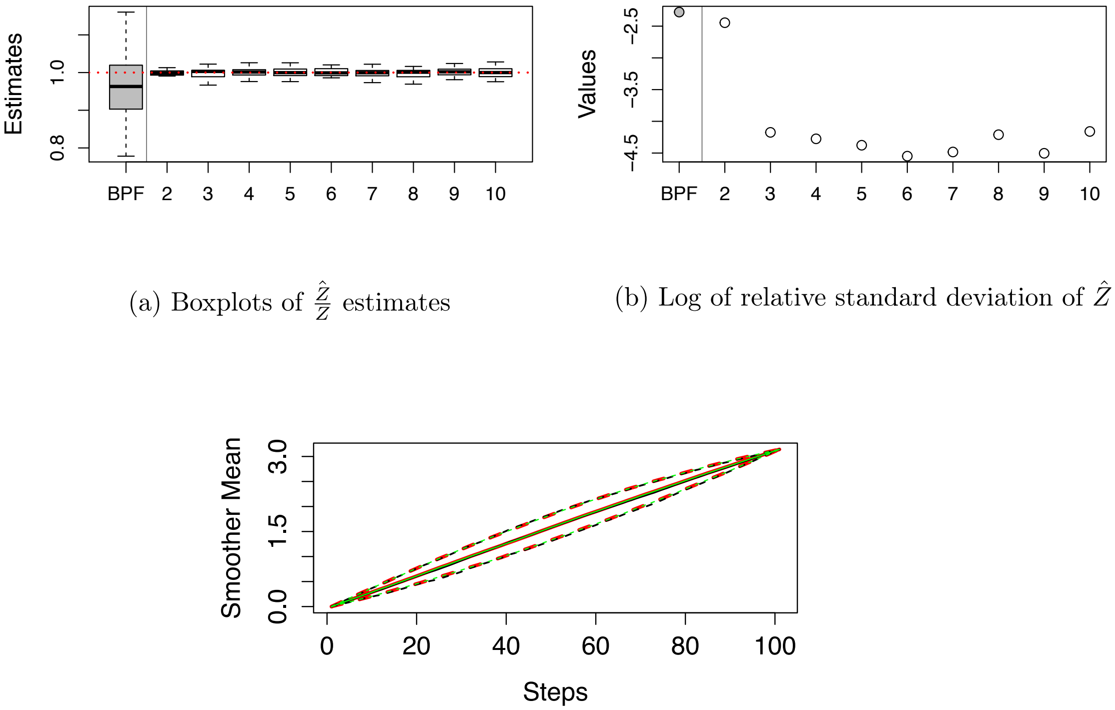


(a) Boxplots of $\frac{\hat{Z}}{Z}$ estimates

(b) Log of relative standard deviation of $\hat{Z}$

(c) Smoother mean for the iAPF (red), BPF (black), EA (green) ±1 smoother standard deviation

Figure 12.6: Simulations with $h = 10^{-1}$, $c = 0$

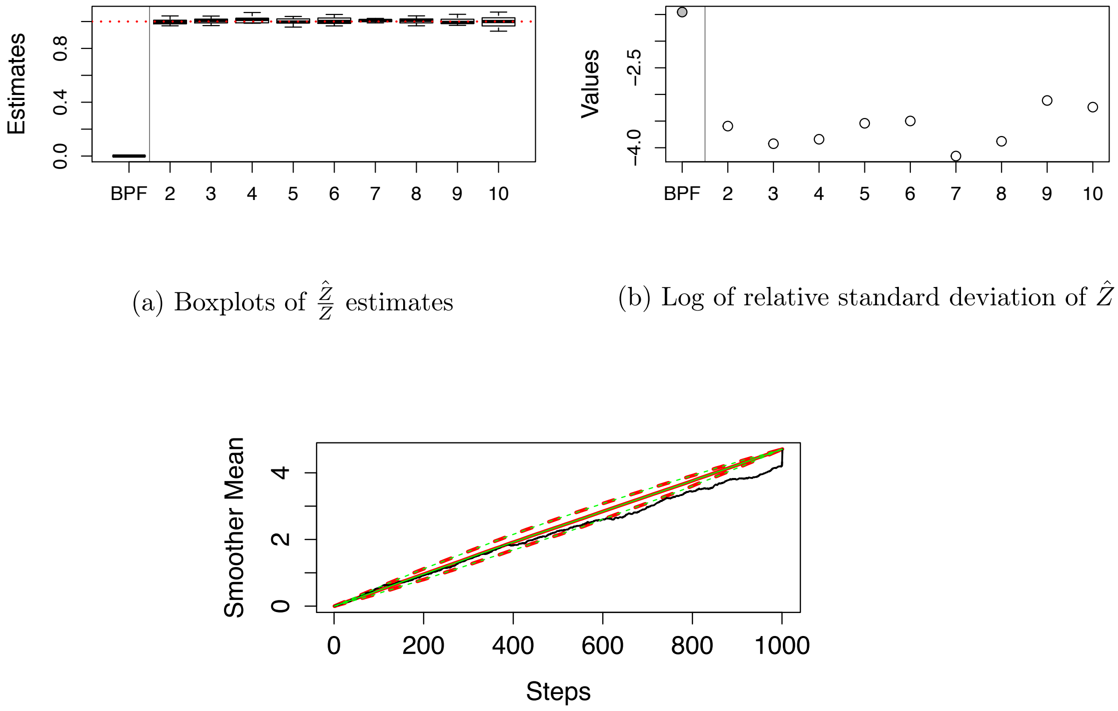


(a) Boxplots of $\frac{\hat{Z}}{Z}$ estimates

(b) Log of relative standard deviation of $\hat{Z}$

(c) Smoother mean for the iAPF (red), BPF (black), EA (green) ±1 smoother standard deviation

Figure 12.7: Simulations with $h = 10^{-2}$, $c = \pi$



(a) Boxplots of $\frac{\hat{Z}}{Z}$ estimates

(b) Log of relative standard deviation of $\hat{Z}$

(c) Smoother mean for the iAPF (red), BPF (black), EA (green) ±1 smoother standard deviation

Figure 12.8: Simulations with $h = 10^{-3}$, $c = \frac{3}{2}\pi$

12.3 Time-dependent drift and diffusion in 2 dimensions

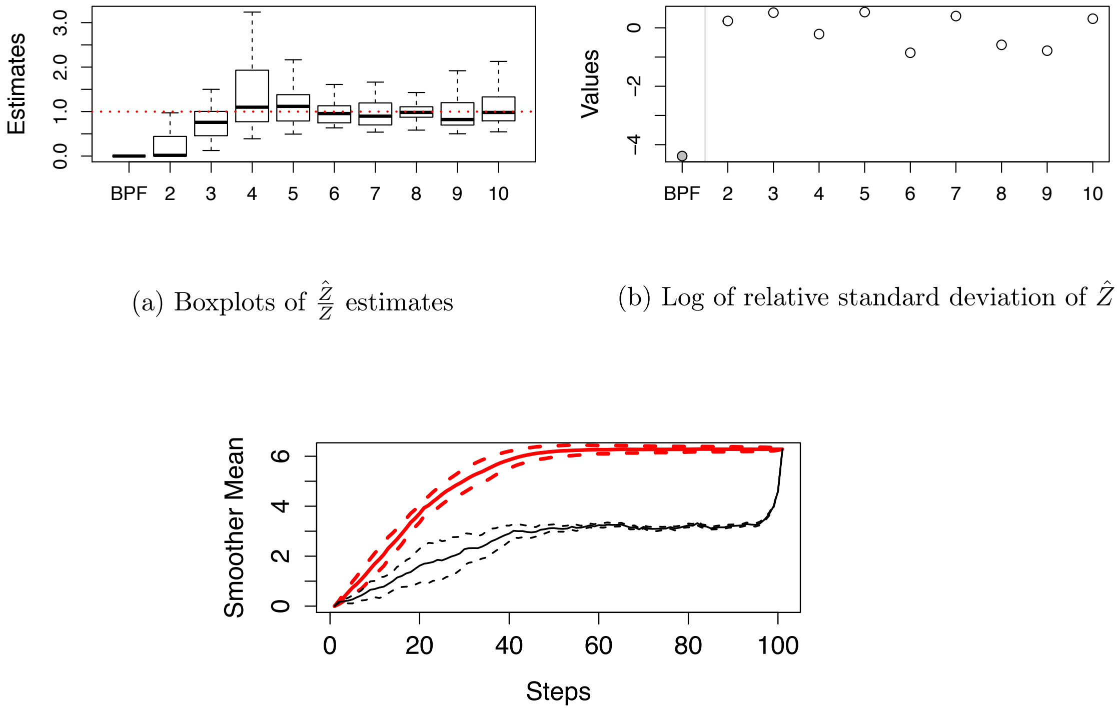
Given a one dimensional diffusion, provided that the drift a and the volatility function b do not depend on time, that they satisfy the regularity conditions (Lipschitz (10.3), with a growth bound (10.2)) and that the conditions to allow the Lamperti transform hold we can obtain exact samples of diffusion bridges through the exact algorithm. We now run some simulations with the iAPF for two more general examples to investigate the behaviour of the algorithm when the Exact Algorithm cannot directly be used. In the first one-dimensional example the drift function depends also on the time parameter and with the second example we consider a diffusion in two dimensions.

12.3.1 Time dependent drift

Consider the diffusion

$$d\bar{X}_s = \alpha s \sin(\bar{X}_s)ds + \beta dW_s$$

with the conditions $\bar{X}_0 = 0$, $\bar{X}_1 = c > 0$ in $[0, 1]$. We run simulations for different parameter configurations to compare the performances of the BPF and the iAPF. In all our simulations the iAPF gives a better performance, the extent of the improved efficiency depending on the instance. The BPF is challenged when the ending point of the diffusion $\bar{X}_1 = c$ is relatively unlikely given the parameters α and β. Consider the parameters $c = 2\pi$, $\alpha = 50$ and $\beta = 3$. We report simulations relative to 50 runs of the BPF with $N = 10000$ particles and iAPF with $N_0 = 500$ particles in Figure 12.9. In this case, because the diffusion paths are continuous and $\alpha, s \geq 0$, the diffusion process has to go through a regime of negative drift $a\left(s, \bar{X}_s\right) = \alpha s \sin\left(\bar{X}_s\right)$ for $\bar{X}_s \in (\pi, 2\pi)$. If this happens at the beginning of the interval $[0, 1]$, the drift is lower in absolute value because of the time coefficient s. The iAPF can recognise this dynamic thanks to the backward recursion that incorporates information about the ending point in the twisted transition. The BPF fails to provide any reasonable estimate.



(a) Boxplots of $\frac{\hat{Z}}{Z}$ estimates

(b) Log of relative standard deviation of $\hat{Z}$

(c) Smoother mean for the iAPF (red), BPF (black)

Figure 12.9: Simulations with $h = 10^{-2}$, $\bar{x}_S = 2\pi$

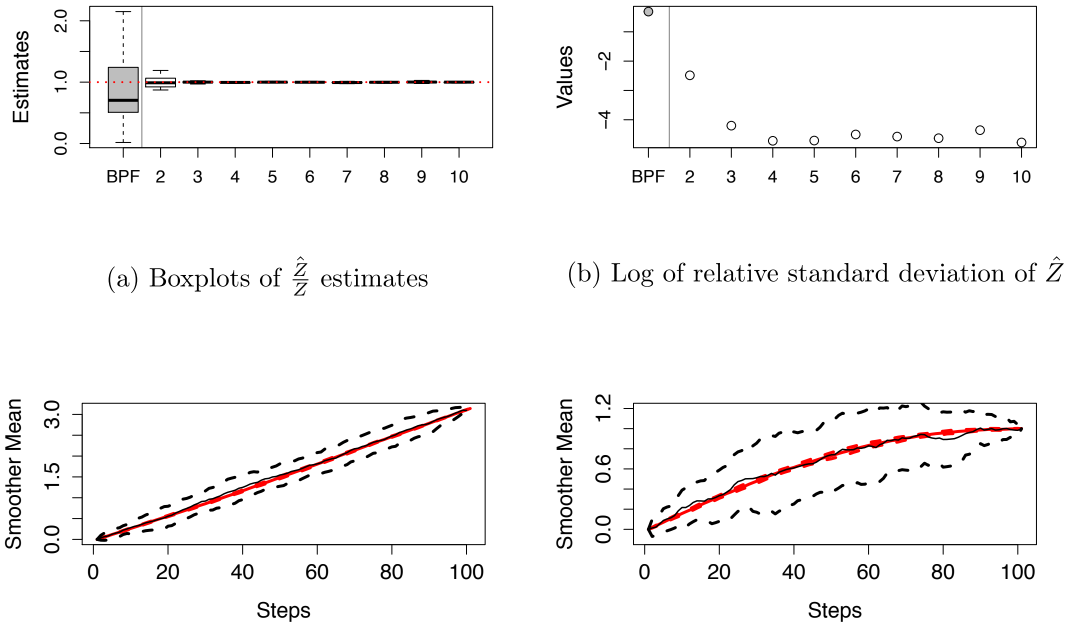
12.3.2 Diffusion in 2 dimensions

Our second example is based on the two dimensional diffusion defined by the stochastic differential equations

$$\begin{aligned} d\bar{X}_s^{(1)} &= \alpha \bar{X}_s^{(2)} \sin(\bar{X}_s^{(1)}) ds + \beta dW_s^{(1)} \\ d\bar{X}_s^{(2)} &= dW_s^{(2)} \end{aligned}$$

with the conditions $\bar{X}_0^{(1)} = \bar{X}_0^{(2)} = 0$, $\bar{X}_1^{(1)} = c$, $\bar{X}_1^{(2)} = 1$ in $[0, 1]$ with two independent Brownian motions $W_s^{(1)}$ and $W_s^{(2)}$. The iAPF presents some improvements for all the investigated parameter configurations. We report simulations relative to 50 runs of the BPF with $N = 10000$ particles and iAPF with $N_0 = 500$ particles relative to two different parameter sets. We set $\alpha = \beta = 1$

and $c = \pi$ and report the results in Figure 12.10. From Figure 12.10a and Figure 12.10b we can see that the iAPF significantly outperforms the BPF, but the relative standard deviation of the estimates from the BPF is still under control. In Figure 12.10c and Figure 12.10d we report the smoothing mean estimated by the two algorithm plus and minus one empirical standard deviation for the 50 replicates. The black (BPF) and the red (iAPF) paths do not differ qualitatively but the standard deviation of the BPF mean estimator (black dashed lines) is significantly higher than that of the iAPF (red dashed lines).



(a) Boxplots of $\frac{\hat{Z}}{Z}$ estimates

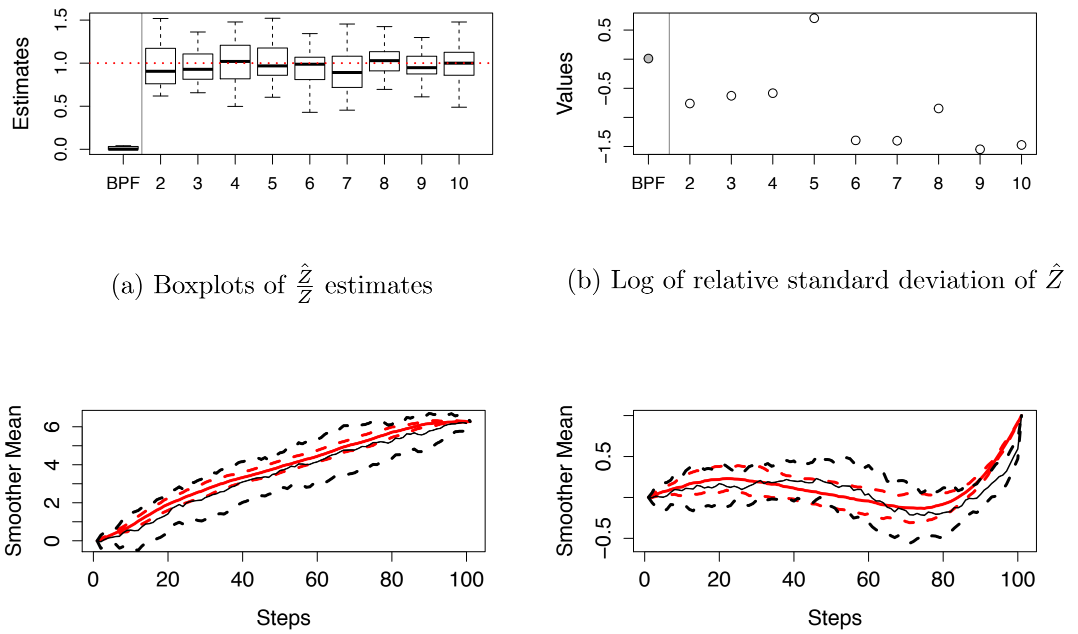
(b) Log of relative standard deviation of $\hat{Z}$

(c) Smoother mean for $X_t^{(1)}$: iAPF (red), BPF (black)

(d) Smoother mean for $X_t^{(2)}$: iAPF (red), BPF (black)

Figure 12.10: Simulations with $h = 10^{-2}$, $c = \pi$

We provide a similar report for the parameters α =50, $\beta = 3$ and $c = 2\pi$ in Figure 12.11. In this second more challenging instance the BPF cannot provide reasonable estimates for the approximation of the transition density Z_h while the relative standard deviation of the estimates provided by the iAPF is well under control.



(a) Boxplots of $\frac{\hat{Z}}{Z}$ estimates

(b) Log of relative standard deviation of $\hat{Z}$

(c) Smoother mean for $X_t^{(1)}$: iAPF (red), BPF (black)

(d) Smoother mean for $X_t^{(2)}$: iAPF (red), BPF (black)

Figure 12.11: Simulations with $h = 10^{-2}$, $c = 2\pi$

12.4 Comparison with other methods

In all the exploratory examples of this chapter so far we have compared the iAPF with the BPF. Applying the BPF to the Euler–Maruyama approximation of a diffusion is not something novel: it is known under the name of forward simulation and it is described in its essence in Pedersen (1995). Also the drawbacks of forward simulation are well known, as when simulating skeleton paths with this method we do so independently of the observations. The modified diffusion bridges (MDB) approach of Durham & Gallant (2002) can overcome this limit by shaping a proposal transition which depends on the ending point $\bar{x}_S$ of the diffusion bridge. This method, like the forward simulation method, is based on the Euler–Maruyama approximation of the diffusion bridge defined by equation 10.1 with discretisation parameter h. When propagating a path $x_{1:t}$ with the MDB approach, we define the approximate diffusion

$$dZ_s = a\left(x_t, th\right) ds + b\left(x_t, th\right) dW_s$$

for $s \in [th, S]$ and with the condition $Z_{th} = x_t$. The subsequent point x_{t+1} of the path $x_{1:t}$ is proposed according to the discretisation of this auxiliary diffusion, conditioning on the ending point $Z_S = \bar{x}_S$. In particular we have that

$$\begin{pmatrix} Z_{h(t+1)} \\ Z_S \end{pmatrix} \mid \left(Z_{th} = x_t\right) \sim \mathcal{N}\left(m\left(x_t, th\right), \Sigma\left(x_t, th\right)\right)$$

where

$$m\left(x_t, th\right) = \begin{pmatrix} x_t + ha\left(x_t, th\right) \\ x_t + \left(S - th\right) a\left(x_t, th\right) \end{pmatrix}$$

and

$$\Sigma\left(x_t, th\right) = \begin{pmatrix} hb(x_t, th)b'\left(x_t, th\right) & hb(x_t, th)b'\left(x_t, th\right) \\ hb(x_t, th)b'\left(x_t, th\right) & \left(S - th\right) b(x_t, th)b'\left(x_t, th\right) \end{pmatrix}.$$

From this we can easily and computationally efficiently derive the conditional distribution of $\left(Z_{h(t+1)} \mid Z_{ht} = x_t, Z_S = \bar{x}_S\right)$ and propose x_{t+1} accordingly (see for example Malory & Sherlock (2016)). Once the full path $x_{1:T}$ is sampled, it is given with the appropriate importance weight as usual. While this approach can be very efficient in many scenarios, paths sampled this way present necessarily a linear dynamic therefore are not suitable for approximating a continuous diffusion with a highly non-linear dynamic, as we will show in the following subsections.

With the aim of addressing this problem for highly non-linear diffusion, a residual-bridge proposal is introduced in Whitaker et al. (2016). This method consists of constructing a deterministic path $(\zeta_s)_{s\in[0,S]}$ which captures the non-linear dynamic of the conditioned diffusion, and then using the MDB approach on the residual process $R_s := \bar{X}_s - \zeta_s$ which satisfies the stochastic differential equation

$$dR_s = \left(a\left(R_s + \zeta_s, s\right) - \zeta'_s\right) ds + b\left(R_s + \zeta_s, s\right) dW_s \tag{12.2}$$

for $s \in [0, S]$ and with the condition $R_0 = 0$. This method is successful if the path $(\zeta_s)_{s\in[0,S]}$ effectively captures the non-linear dynamic of the true diffusion so that the residual process resembles a Brownian bridge. The choice of the deterministic path $(\zeta_s)_{s\in[0,S]}$ is critical. A natural choice (see Whitaker et al. (2016)) is to define $(\zeta_s)_{s\in[0,S]} = (\eta_s)_{s\in[0,S]}$, where $(\eta_s)_{s\in[0,S]}$ is the solution of the ordinary differential equation

$$\frac{d\eta_s}{ds} = a\left(\eta_s, s\right) \tag{12.3}$$

with the condition $\eta_0 = \bar{x}_0$, that is obtained by suppressing the stochastic term in the stochastic differential equation in 10.1. Such deterministic path does not take into account the observation $\bar{X}_S = \bar{x}_S$ and therefore can be inconsistent with the conditioned diffusion. Another suggestion also presented in Whitaker et al. (2016) is to consider a tractable approximation $\left(\hat{R}_s\right)_{s\in[0,S]}$ of the process $(R_s)_{s\in[0,S]}$ and define the deterministic path as

$$\zeta_s = \eta_s + \mathbb{E}\left[\hat{R}_s \mid \bar{X}_S = \bar{x}_S\right]$$

where $(\eta_s)_{s\in[0,S]}$ is again the solution of Equation (12.3).

An important extension of the residual-bridge work of Whitaker et al. (2016) is presented in Malory & Sherlock (2016). In particular their proposal takes into account diffusion volatilities which are not constant. This can lead to greater statistical efficiency in situations where the volatility varies considerably such as for large inter-observation intervals where the diffusion moves substantially over the state space and for diffusions whose volatility is time inhomogeneous.

12.4.1 Sin diffusion

We consider the sin diffusion

$$dX_t = \alpha \sin\left(X_t\right) dt + dW_t \tag{12.4}$$

in $[0, 1]$ with condition $X_0 = 0$, $X_1 = c$, the parameter $\alpha > 0$ that controls the magnitude of the drift and the brownian motion W_t. We initially set the parameter $\alpha = 1$ and we repeat the first experiment as in Subsection 12.2 using the MDB approach of Durham & Gallant (2002). As we did before, we run simulations for all the combinations of values for the parameters $c \in \left\{0, \pi, \frac{3}{2}\pi\right\}$ and discretisation parameter $h \in \left\{10^{-1}, 10^{-2}, 10^{-3}\right\}$ and we report the updated version of Table 12.1 in Table 12.3. Recall that each entry of the table consists of the standard deviations of $\frac{\hat{Z}}{Z}$ for 50 iterations of the iAPF with $N_0 = 500$, the BPF with $N = 10000$ and the MDB approach with $N = 10000$. The number of particles in use for each algorithm is chosen so that they lead to a similar computational time. We can see in Table 12.3 that for $\alpha = 1$ in every instance the MDB approach shows the best accuracy as measured by the relative variance of the normalising constant estimates.

Table 12.2: Logarithm of standard deviation of $\frac{\hat{Z}}{Z}$ for iAPF, BPF, D&G

$\alpha = 1$	$h = 10^{-1}$	$h = 10^{-2}$	$h = 10^{-3}$
$\bar{x}_S = 0$	iAPF: $-$ 5.58 BPF: $-$ 4.21 D&G: $-$ 5.86	iAPF: $-$ 5.70 BPF: $-$ 3.42 D&G: $-$ 7.18	iAPF: $-$ 5.91 BPF: $-$ 2.98 D&G: $-$ 6.78
$\bar{x}_S = \pi$	iAPF: $-$ 4.48 BPF: $-$ 3.34 D&G: $-$ 6.51	iAPF: $-$ 4.14 BPF: $-$ 2.38 D&G: $-$ 6.71	iAPF: $-$ 4.59 BPF: $-$ 1.49 D&G: $-$ 6.78
$\bar{x}_S = \frac{3}{2}\pi$	iAPF: $-$ 3.92 BPF: $+$ 0.45 D&G: $-$ 6.20	iAPF: $-$ 4.22 BPF: $-$ D&G: $-$ 7.08	iAPF: $-$ 3.28 BPF: $-$ D&G: $-$ 6.92

Now we run the same experiment setting the parameter α that controls the drift to $\alpha = 10$, and we report the results from these simulations in Table 12.3. Even in this more challenging scenario the MDB approach leads to an acceptable variance of the normalising constant estimates, while the estimates from the BPF are extremely inaccurate apart from the instance with $(c, h) = (0, 10^{-1})$. Nonetheless the iAPF presents the best performance among the three methods, and in particular for $c \in \left\{\pi, \frac{3}{2}\pi\right\}$ the relative variance of the estimates is at least one order of magnitude lower than that of the MDB approach for every instance. As we pick more extreme values for the parameter α we accentuate the non-linearity of the drift and for this reason the MDB approach becomes unsuitable for example for the problem of evaluating transition densities for this model, as we show with the following experiment.

Table 12.3: Logarithm of standard deviation of $\frac{\hat{Z}}{Z}$ for iAPF, BPF, D&G

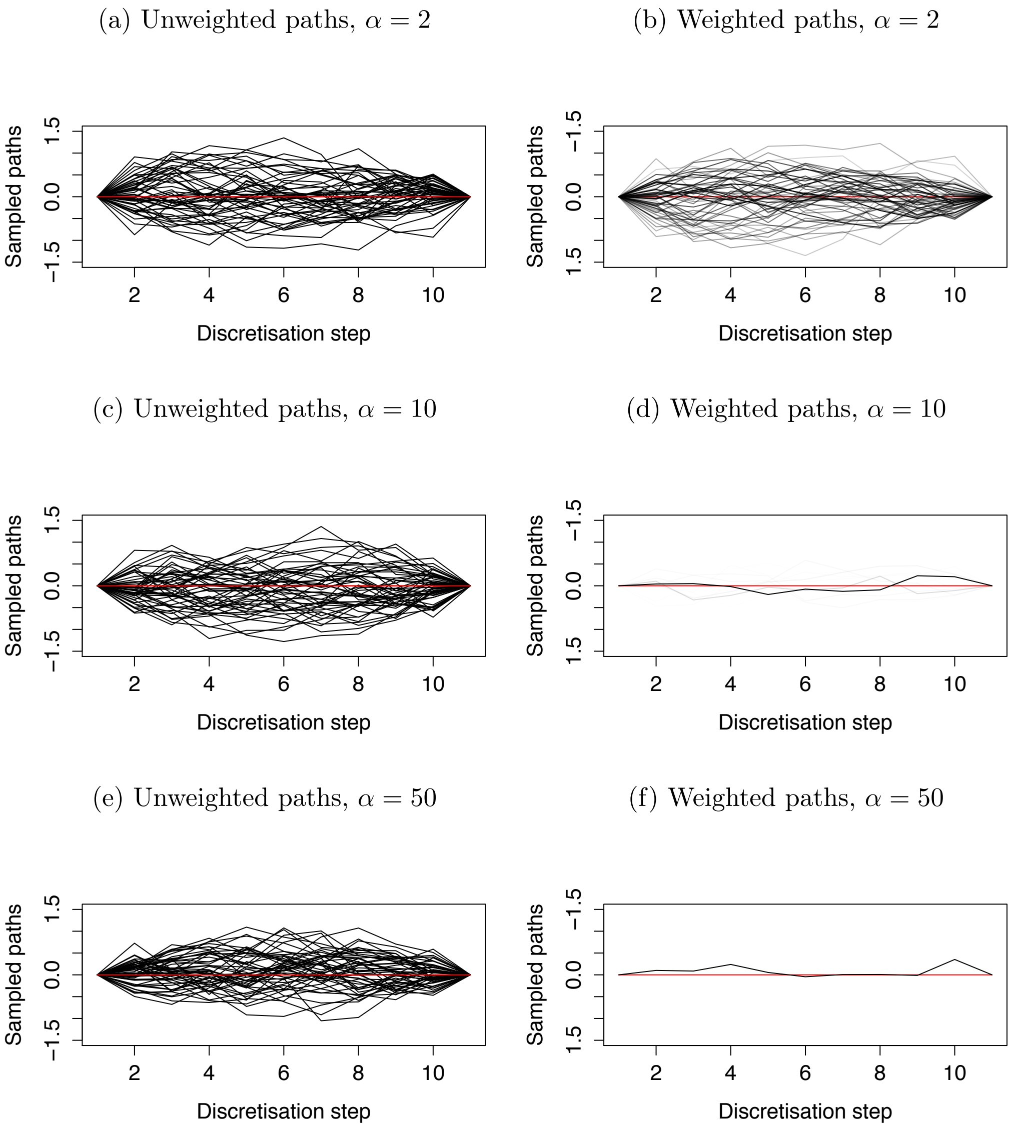
$\alpha = 10$	$h = 10^{-1}$	$h = 10^{-2}$	$h = 10^{-3}$
$\bar{x}_S = 0$	iAPF: $-$ 5.79 BPF: $-$ 2.91 D&G: $-$ 3.15	iAPF: $-$ 5.24 BPF: $-$ 0.26 D&G: $-$ 7.18	iAPF: $-$ 5.01 BPF: $+$ 0.46 D&G: $-$ 6.78
$\bar{x}_S = \pi$	iAPF: $-$ 3.08 BPF: $-$ 0.36 D&G: $-$ 0.78	iAPF: $-$ 3.53 BPF: $+$ 1.14 D&G: $-$ 0.52	iAPF: $-$ 3.19 BPF:$-$ D&G: $-$ 0.58
$\bar{x}_S = \frac{3}{2}\pi$	iAPF: $-$ 2.80 BPF: $+$ 0.45 D&G: $-$ 0.50	iAPF: $-$ 2.22 BPF: $-$ D&G: $-$ 0.90	iAPF: $-$ 2.28 BPF: $-$ D&G: $-$ 1.08

We fix the right end of the bridge to $c = 0$ and the discretisation parameter to $h = 10^{-1}$, and we use the MDB approach to produce 50 sample paths for different value of the parameter $\alpha \in \{2, 10, 50\}$. We present the results in Figure 12.12. For each case the subfigure on the left shows the 50 sample paths in black, and the subfigure on the right reports the same paths but the transparency factor of each one is proportional to its importance weight. While the MDB proposal seems suitable for the case where $\alpha = 2$, when we set $\alpha = 10$ or $\alpha = 50$ the degeneracy of the importance weights is evident as in these cases only one path is clearly visible. For these cases the iAPF is a better choice: after each time step, once a twisted model is defined through the forward backward procedure, the resampling feature allows us to drop the paths that are doomed to have a negligible importance weight, but this is not the only reason. With the forward backward procedure the iAPF defines a sequence of twisted proposals that take into account the ending point of the bridge but also adapt to the non-linear drift of the diffusion. To show this, for the diffusion in 12.4 with parameters $c = 0$ and $\alpha \in \{2, 10, 50\}$ we run an iAPF with $N = 50$ particles, keeping the number of particles constant for 10 iterations of the algorithm in order to have a fixed computational cost of order $\mathcal{O}(N)$. At the end of the 10th iteration we store the final sequence $\boldsymbol{\psi}$ of look-ahead functions and we run a $\boldsymbol{\psi}$-APF without resampling. We report the results in Figure 12.13 with the same interpretation as in Figure 12.12. We also report in Table 12.4 the effective sample size of the 6 sets of 50 weights of the paths sampled with the MDB approach and with the $\boldsymbol{\psi}$-APF. Recall that the effective sample size n_{eff} Kong et al. (1994) of a weighted sample $\{x_i, w_i\}_{i \in 1:N}$ is given by

$$n_{eff} = \frac{\sum_{i=1}^{N} w_i^2}{\left(\sum_{i=1}^{N} w_i\right)^2}$$

so for any set of N weights we have $1 \leq n_{eff} \leq N$. The effective sample size gives the approximate size of an i.i.d. sample which would be equivalent in precision to the weighted sample used in a single importance sampling setting. For the case $\alpha = 50$ the effective sample size of the weights from MDB approach reach the lower limit for this indicator. We repeat the identical experiment with the number of particles that we used for the other simulations: 10000 for the MDB approach and 500 for the $\boldsymbol{\psi}$-APF. The results are reported in Table 12.5.

Figure 12.12: MDB approach for the sin diffusion: 50 sampled paths



(a) Unweighted paths, $\alpha = 2$

(b) Weighted paths, $\alpha = 2$

(c) Unweighted paths, $\alpha = 10$

(d) Weighted paths, $\alpha = 10$

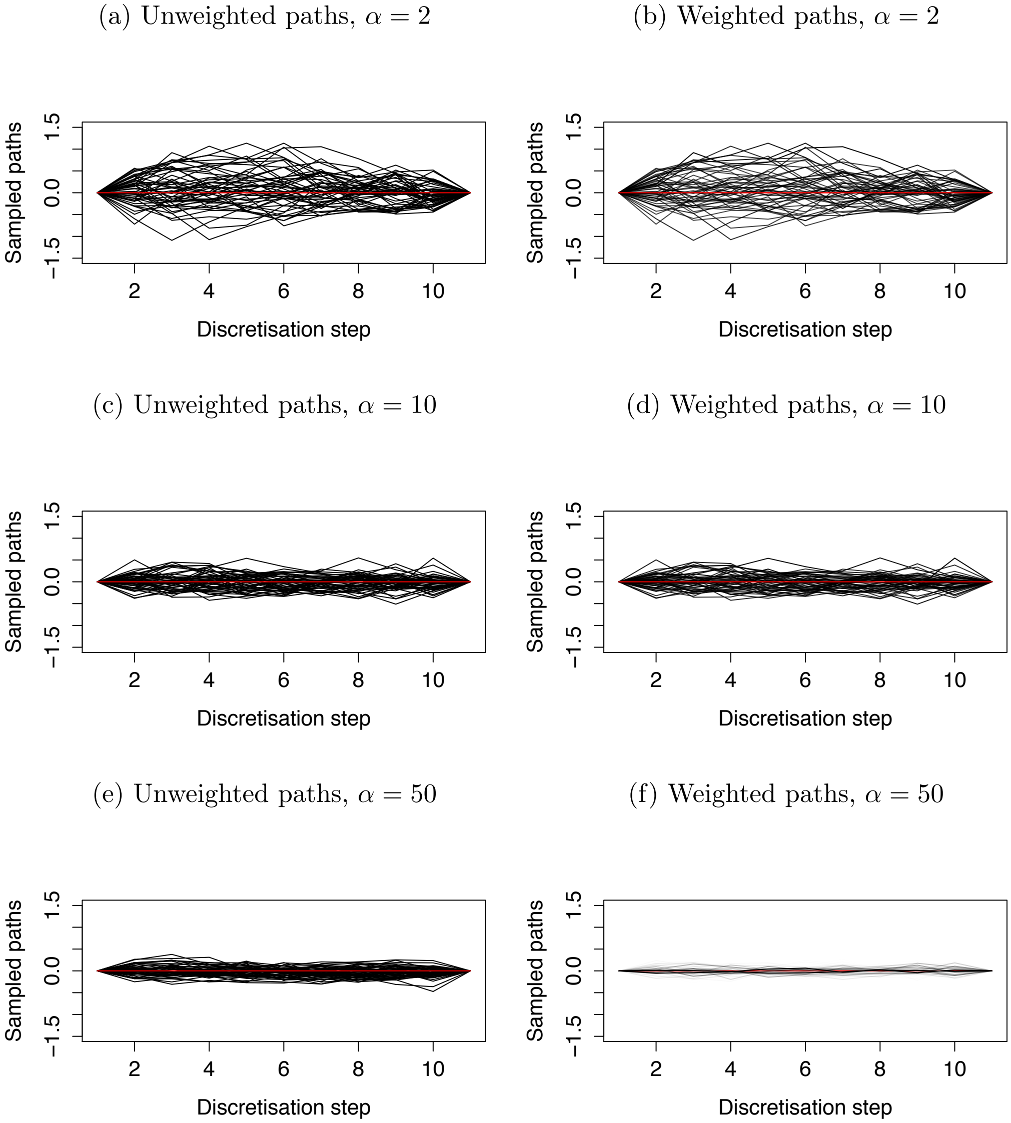
(e) Unweighted paths, $\alpha = 50$

(f) Weighted paths, $\alpha = 50$

Table 12.4: Effective sample size of the weights of the path in Figure 12.12 and Figure 12.13

ESS	MDB approach	$\boldsymbol{\psi}$-APF
$\alpha = 2$	41.02	49.86
$\alpha = 10$	2.23	49.92
$\alpha = 50$	1.00	6.78

Figure 12.13: iAPF for the sin diffusion: 50 sampled paths



(a) Unweighted paths, $\alpha = 2$

(b) Weighted paths, $\alpha = 2$

(c) Unweighted paths, $\alpha = 10$

(d) Weighted paths, $\alpha = 10$

(e) Unweighted paths, $\alpha = 50$

(f) Weighted paths, $\alpha = 50$

For this case the extensions of the MDB approach presented in Section 12.4

Table 12.5: Effective sample size from sampled paths from the MDB approach (10000 paths) and the $\boldsymbol{\psi}$-APF without resampling (500 paths).

ESS	MDB approach	$\boldsymbol{\psi}$-APF
$\alpha = 2$	8324.07	495.33
$\alpha = 10$	406.12	490.74
$\alpha = 50$	1.08	64.52

cannot mitigate the problem related to non-linear drift. Consider the residual-bridge approach of Whitaker et al. (2016). If we consider the natural choice of choosing a deterministic path capturing the non-linearity of the drift, the ordinary differential equation 12.3 becomes in this case

$$\frac{d\eta_s}{ds} = \alpha \sin \eta_s$$

with the condition $\eta_0 = 0$. This equation presents a unique solution which is $\eta_s \equiv 0$, therefore applying the MDB approach to the residual-bridge defined by the stochastic differential equation 12.2 is equivalent to applying it to the original diffusion. Also the other method suggested by Whitaker et al. (2016) based on a tractable process $\hat{R}_s$ approximating the residual reduces to the simple MDB approach when the process $\bar{X}_s$ is observed perfectly and $\eta_S = \bar{x}_S$. Not even the other extension we considered by Malory & Sherlock (2016) can enhance the performance of the MDB approach, as this is designed to address scenarios where the volatility varies substantially, while in this case the volatility of the process is constant in time and space.

12.4.2 Stochastic volatility process of the Heston model

The Heston model is a mathematical continuous time stochastic model popular in the financial literature that describes the evolution of the prices of a set of financial assets that depend on a common underlying volatility diffusion (Heston (1993)). We usually observe the prices of the assets from the financial market at discrete times and we want to make inference on the parameters guiding the stochastic volatility process. We examine some instances of this problem in the next section, where it is difficult to adapt the MDB approach of Durham & Gallant (2002) as perfect observations of the volatility process are not available. In this subsection we make a comparison between the iAPF and the MDB approach for the simplified model that consists only in the underlying stochastic volatility process.

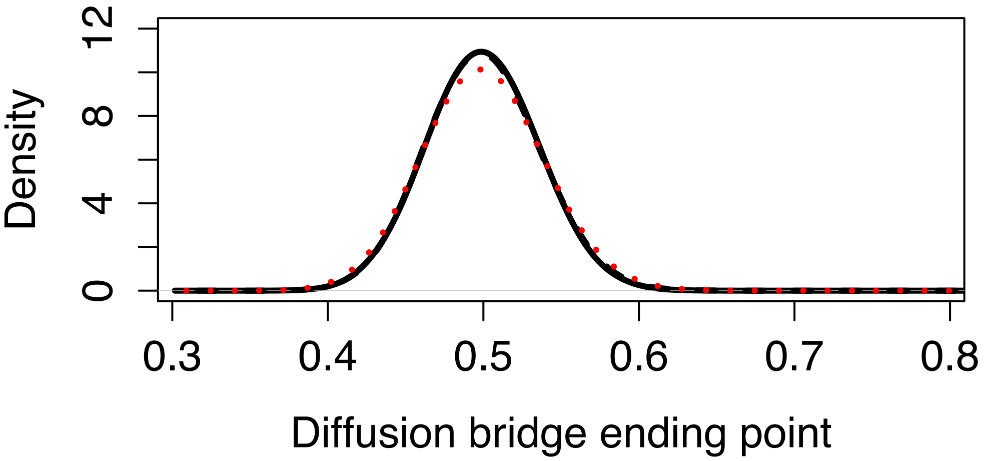
Consider volatility process v_t following the stochastic differential equation

$$dv_t = \theta_1 \left(\theta_2 - v_t\right) dt + \theta_3 \sqrt{v_t} dW_t \tag{12.5}$$

for $t \in [0,1]$ and the Brownian motion W_t. This corresponds to the Cox-Ingersoll-Ross model for interest rates (Cox et al. (1985)) where $\theta_2 > 0$ is the long run average variance, $\theta_1 > 0$ the rate at which v_t reverts to θ_2 and θ_3 is the volatility of the volatility parameter. For this model we consider the problem of estimating the transition density $\mathsf{p}_{0,1}\left(\bar{v}_0, \cdot\right)$ for a fixed $\bar{v}_0 > 0$.

We fix the parameters $\theta_2 = 0.5$, $\theta_3 = 0.3$ and $\bar{v}_0 = 0.5$ and we investigate the three different cases for $\theta_1 \in \{5, 10, 20\}$. We consider a set of equally spaced points $v_1^i = 0.3 + i\frac{0.8-0.3}{100}$ for $i \in 1:100$ in the interval $[0.3, 0.8]$. For each v_1^i our estimate for $\mathsf{p}_{0,1}\left(\bar{v}_0, v_1^i\right)$ is the normalising constant Z_h of the SSM given by the Euler–Maruyama discretisation of the diffusion (12.5) corresponding to the discretisation parameter h. In order to set a parameter h such that the corresponding discretisation is fine enough for our estimates to be close to the true transition density $\mathsf{p}_{0,1}\left(\bar{v}_0, \cdot\right)$ we run some preliminary simulations. For $\theta_1 = 20$ and for $i \in 0:100$ we use the iAPF with $N_0 = 1000$ starting particles to compute an estimate of $\mathsf{p}_{0,1}\left(\bar{v}_0, v_1^i\right)$. We do so for three different values of the discretisation parameter $h \in \{50, 100, 200\}$ and we report the three smoothed estimates of the density $\mathsf{p}_{0,1}\left(\bar{v}_0, \cdot\right)$ in Figure 12.14.

Figure 12.14: Estimates of $\mathsf{p}_{0,1}(\bar{v}_0, \cdot)$ given by the parameters $h = 50$ (red, dotted), $h = 100$ (black, dashed) and $h = 200$ (black,solid)

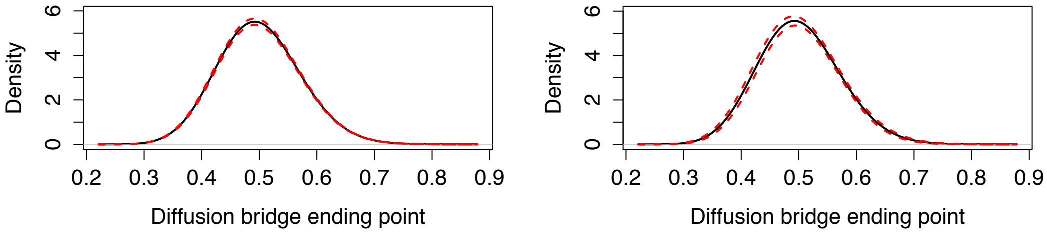


While we can see a small difference between the estimates from $h = 50$ and $h = 100$, the two estimates obtained setting $h = 100$ and $h = 200$ are practically overlapping. This suggests that these approximations should be reasonably close to the transition density of the continuous model. For this reason we set $h = 200$ for the rest of this section.

In order to assess the variability of the estimated transition density, we run a similar experiment but this time for each v_1^i we get 50 estimates from 50 runs of the algorithm and we register the mean and the standard deviation of these estimates. We do so using an iAPF with $N_0 = 500$ starting particles and then with the MDB approach of Durham & Gallant (2002), using $N = 10000$, leading to a comparable (but lower for the iAPF) computational time with respect to the iAPF. In Figure 12.15 we report the results for the instance of the model with $\theta_1 = 5$. The black solid line is obtained by smoothing the means of the estimates for the different values $\{v_1^i\}_{i \in 0:100}$, while the red dashed lines represent the variability of the densities obtained by adding and subtracting two estimated standard deviations. For this first case, with similar computational times the performances of the iAPF and the MDB approach are very similar, with this second algorithm offering a slightly lower variance of the estimates.

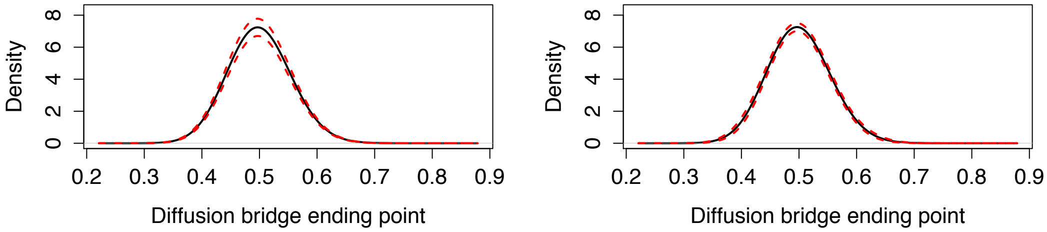
As we increase the parameter θ_1 that controls the drift of the diffusion, the accuracy of the MDB gets rapidly worse while the iAPF is not visibly effected, similarly to the example in the previous section. In particular for $\theta_1 = 10$ in Figure 12.16 the iAPF overtakes the MDB approach in accuracy only slightly, while for the value $\theta_1 = 20$ in Figure 12.17 the difference between the two estimates is remarkable.

Figure 12.15: Estimates of transition density $\mathsf{p}_{0,1}(\bar{v}_0, \cdot)$ for $\theta_1 = 5$


(a) MDB approach

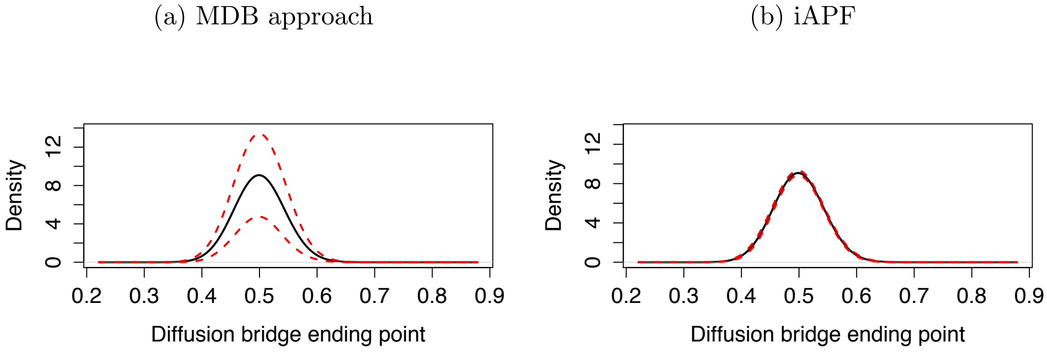
(b) iAPF

Figure 12.16: Estimates of transition density $\mathsf{p}_{0,1}(\bar{v}_0, \cdot)$ for $\theta_1 = 10$


(a) MDB approach

(b) iAPF

Figure 12.17: Estimates of transition density $\mathsf{p}_{0,1}(\bar{v}_0, \cdot)$ for $\theta_1 = 20$


(a) MDB approach

(b) iAPF

12.5 Multivariate Heston model

In this section we consider the problem of estimating the likelihood of a sequence of noisy, discrete and partial observations from a multivariate Heston model (Heston (1993)). The Heston model consists of an underlying volatility process v_t and a set of d financial securities $S^{(1)}, \ldots, S^{(d)}$. The underlying volatility process v_t follows the stochastic differential equation

$$dv_t = \theta_1 (\theta_2 - v_t)\, dt + \theta_3 \sqrt{v_t} dW_t$$

for $t \in [0,1]$ and the Brownian motion W_t, and it is the same as the process 12.5 in Section 12.4.2. The evolution of the values of d assets $S^{(1)}, \ldots, S^{(d)}$ depends on the common underlying volatility process v_t. Each asset $S_t^{(i)}$ follows the stochastic differential equation

$$dS_t^{(i)} = \mu_i S_t^{(i)} dt + S_t^{(i)} \sqrt{v_t} dW_t^{(i)}$$

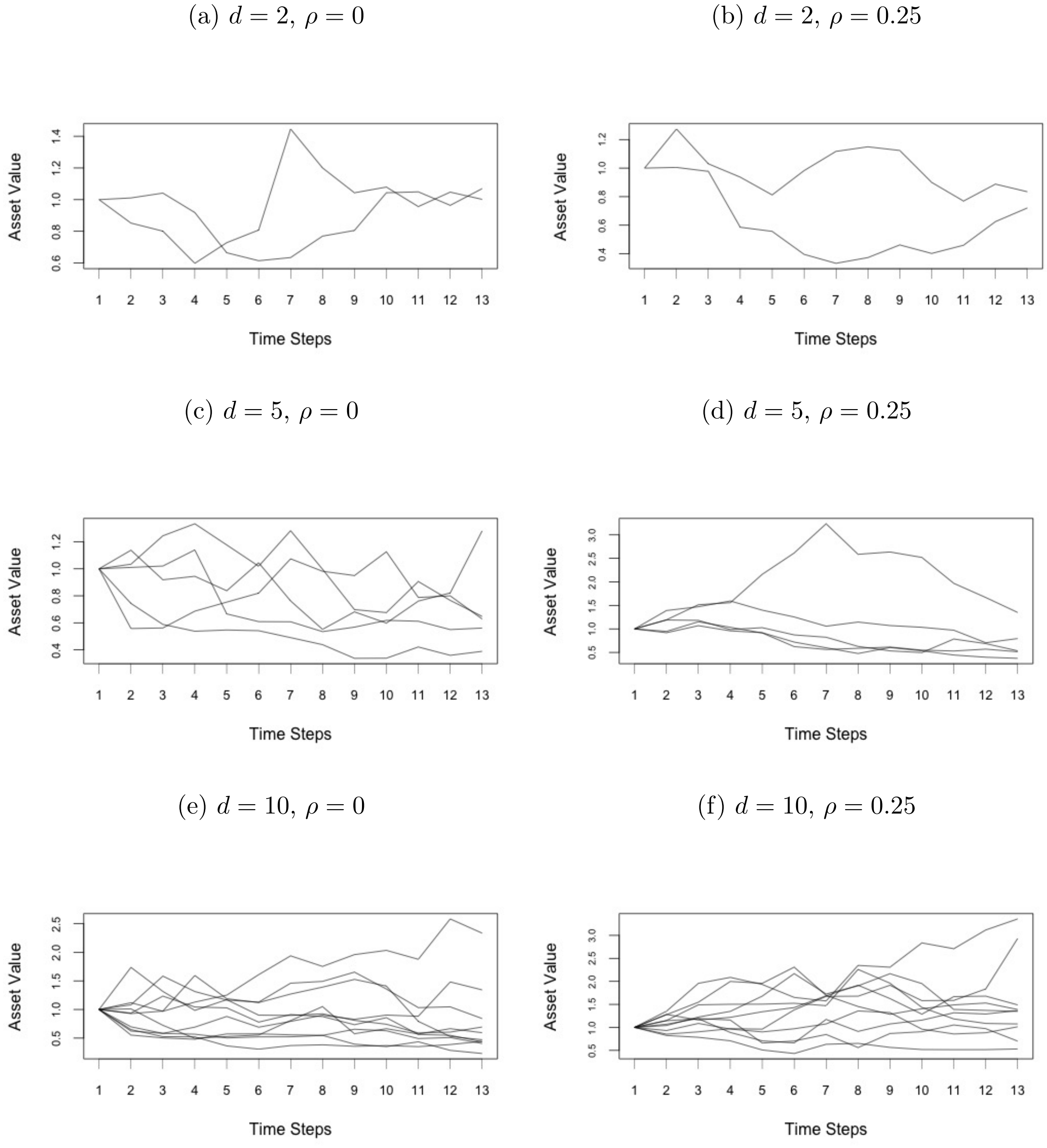
for $t \in [0,1]$ and the possibly correlated Brownian motions $W_t^{(1)}, \ldots, W_t^{(d)}$. While we can read assets prices from the financial market and interpret them as noisy observations of the assets values, we do not have access to any direct observation of the underlying volatility process. Suppose we have $T = 13$ equally spaced noisy observations of each asset (but not of the underlying stochastic volatility process), that represent monthly measurements over the course of a year at times $t_1 = 0$, $t_2 = \frac{1}{12}$, $\ldots$, $t_{13} = 1$. In particular we have

$$y_j^{(i)} \sim \mathcal{N}\left(S_{t_j}^{(i)}, \epsilon\right)$$

for $i \in 1:d$ and $j \in 1:T$. We write $y_j = \left(y_j^{(1)}, \ldots, y_j^{(d)}\right)$ and we want to estimate $Z = p\left(y_{1:T}\right)$. We do so by running the BPF, the MDB approach of Durham & Gallant (2002) and the iAPF for the SSM given by the Euler–Maruyama discretisation of the Heston model diffusion corresponding to the discretisation step $h = 100$. In the following simulations we look at the variability of the estimators of Z_h^N given by the BPF, the MDB approach and the iAPF. As stated before, to contain the variance of such normalising constant estimators is very important, for example when they are used within a PMMH scheme for the parameter estimation problem. Note that we only observe the stochastic volatility process through the asset values process: observations of the full latent process $X_t = \left(v_t, S_t^{(1)}, \ldots, S_t^{(d)}\right)$ are only partial and therefore, in the MDB approach, while the asset prices simulated paths between subsequent observations evolve according to modified diffusion bridges, the unobserved volatility process paths between observations follow a forward simulation type evolution.

For the simulations we fixed the parameters $v_0 = 0.5$, $\theta_1 = 1$, $\theta_2 = 0.5$, $\theta_3 = 0.02$ and $S_0^{(i)} = 1$, $\mu_i = 0.05$ for all $i \in 1:d$. For $i \neq j$ we fix the correlations of the Brownian motions $\operatorname{corr}\left(W_t^{(i)}, W_t^{(j)}\right) = \rho$ if $|i-j| = 1$ and $\operatorname{corr}\left(W_t^{(i)}, W_t^{(j)}\right) = 0$ otherwise. First we produce synthetic asset values path from the Euler–Maruyama approximation of the model corresponding to the discretisation parameter $h = 10000$, for all possible combinations of the parameters $d \in \{2, 5, 10\}$ and $\rho \in \{0, 0.25\}$. In Figure 12.18 we report the simulated data that corresponds to perfectly observed monthly measurements of the asset values.

Figure 12.18: Monthly evolutions of the asset values over the course of a year with $d \in \{2, 5, 10\}$ and $\rho \in \{0, 0.25\}$


(a) $d = 2$, $\rho = 0$

(b) $d = 2$, $\rho = 0.25$

(c) $d = 5$, $\rho = 0$

(d) $d = 5$, $\rho = 0.25$

(e) $d = 10$, $\rho = 0$

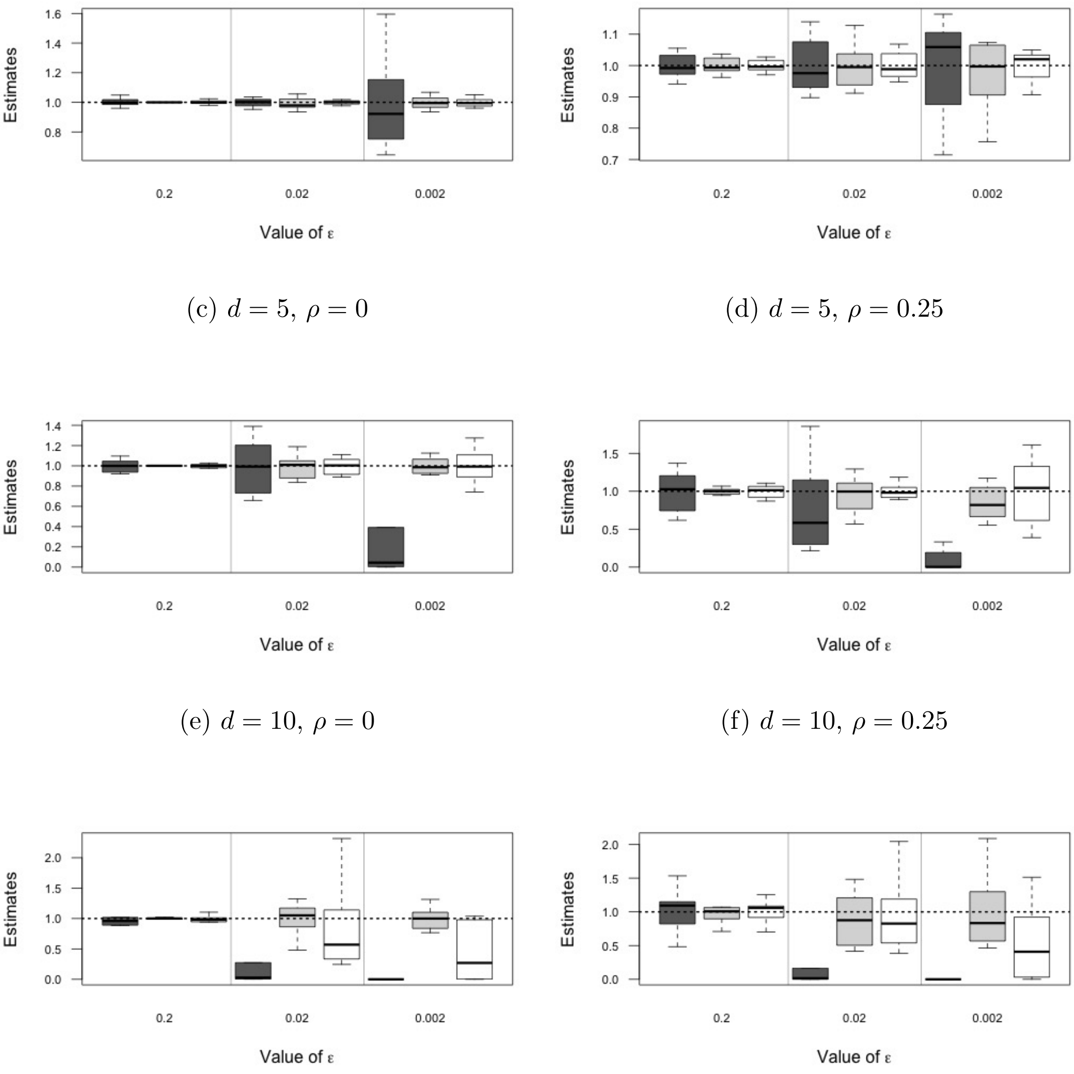
(f) $d = 10$, $\rho = 0.25$

For all the paths in Figure 12.18 we simulate noisy observations for the three value of the parameter $\epsilon \in \{0.2, 0.02, 0.002\}$. For each configuration we run 50 BPF with $N = 10000$ particles and 50 iAPF with $N_0 = 100$ starting particles and we store the corresponding normalising constant estimates. For the MDB approach we set $N_2 = 5000$ particles for the case when $d = 2$, $N_5 = 2500$

particles in the example with $d = 5$ and $N_{10} = 1000$ particles for the final case with $d = 10$. The number of particles is set so that in the worst case the computational time of the iAPF is still lower than that of the other schemes. For each parameter configuration we report in Figure 12.19 the boxplot of the normalising constant estimates divided by the mean of the normalising constant estimates given by the iAPF for that particular configuration, which we believe is the most accurate estimate of $Z = p\left(y_{1:T}\right)$. From the boxplots we can see that in all the instances but one (corresponding to the parameters $d = 2$, $\epsilon = 0.02$ and $\rho = 0$) the estimates from the iAPF present less variability than those obtained from the BPF. The differences in these two algorithms (iAPF and BPF) performances are extremely evident in the most challenging scenarios with $d \geq 5$ and $\epsilon \leq 0.02$. While for most of these cases the BPF estimates are completely unreliable, the iAPF still provides reasonable estimates at a feasible computational time. If compared with the MDB approach, the performance of the iAPF is only clearly superior with respect to that of this scheme in dimension $d = 10$. However in general scenarios the MDB approach presents some important drawbacks with respect to the iAPF. First of all it is necessary for the observation densities to be Gaussian for the implementation of the MDB scheme, whereas the iAPF does not rely on this assumption. Second the MDB approach involves $d \times d$-matrices inversions, therefore the scheme becomes increasingly expensive in higher dimension, gradually reducing its computational cost per particle advantage with respect to the iAPF. Lastly the iAPF easily and automatically adapts to highly non linear dynamics, while the MDB approach fails in this.

In Table 12.6 we also provide all the figures for the logarithms of the relative standard deviations corresponding to the boxplots of Figure 12.19. If we keep d and ϵ constant and we compare the iAPF figures corresponding to the different values of correlation $\rho = 0$ and for $\rho = 0.25$ we notice that the second value is always higher: the iAPF performance is overall inferior for the case with non zero correlation between the Brownian motions ($\rho = 0.25$) with respect to the case with uncorrelated Brownian motions. The reason for this is that the correlation parameter ρ has also a strong influence on the corresponding optimal look-ahead functions. Recall that in all our simulations with the parametric approach we choose the look-ahead functions from the class of Gaussian densities with diagonal variance/covariance matrix (plus a constant). While the iAPF performs still reasonably well with the case $\rho = 0.25$, for higher values of the parameter ρ it is advisable to consider a richer class of Gaussian look-ahead functions, with a variance/covariance matrix with more degrees of freedom (i.e. at least some positive non diagonal entries), in order to capture the dependence structure of the latent process.

Figure 12.19: Boxplots of the normalising constant estimates $\frac{\hat{Z}}{Z}$ for the BPF (dark grey) the iAPF (light grey) and the MDB approach (white)



(a) $d = 2$, $\rho = 0$

(b) $d = 2$, $\rho = 0.25$

(c) $d = 5$, $\rho = 0$

(d) $d = 5$, $\rho = 0.25$

(e) $d = 10$, $\rho = 0$

(f) $d = 10$, $\rho = 0.25$

Table 12.6: Logarithm of standard deviation of $\frac{\hat{Z}}{Z}$ for iAPF, BPF

(a) $\rho = 0$

	$\epsilon = 0.2$	$\epsilon = 0.02$	$\epsilon = 0.002$
$d = 2$	iAPF: $-$ 7.40 BPF: $-$ 3.61 MDB: $-$ 4.39	iAPF: $-$ 2.45 BPF: $-$ 2.95 MDB: $-$ 3.64	iAPF: $-$ 2.77 BPF: $-$ 1.22 MDB: $-$ 3.56
$d = 5$	iAPF: $-$ 5.88 BPF: $-$ 2.85 MDB: $-$ 4.09	iAPF: $-$ 1.78 BPF: $-$ 1.38 MDB: $-$ 3.56	iAPF: $-$ 2.22 BPF: $-$ MDB: $-$ 1.71
$d = 10$	iAPF: $-$ 4.21 BPF: $-$ 1.99 MDB: $-$ 2.27	iAPF: $-$ 1.22 BPF: $-$ MDB: $-$ 0.07	iAPF: $-$ 1.24 BPF: $-$ MDB: $+$ 0.77

(b) $\rho = 0.25$

	$\epsilon = 0.2$	$\epsilon = 0.02$	$\epsilon = 0.002$
$d = 2$	iAPF: $-$ 3.71 BPF: $-$ 3.26 MDB: $-$ 3.96	iAPF: $-$ 2.33 BPF: $-$ 2.47 MDB: $-$ 3.16	iAPF: $-$ 1.85 BPF: $-$ 1.53 MDB: $-$ 2.70
$d = 5$	iAPF: $-$ 2.87 BPF: $-$ 1.36 MDB: $-$ 2.54	iAPF: $-$ 0.72 BPF: $+$ 0.09 MDB: $-$ 1.97	iAPF: $-$ 0.17 BPF: $-$ MDB: $-$ 0.88
$d = 10$	iAPF: $-$ 1.20 BPF: $-$ 1.17 MDB: $-$ 1.11	iAPF: $-$ 0.19 BPF: $-$ MDB: $-$ 0.04	iAPF: $-$ 0.09 BPF: $-$ MDB: $+$ 0.56

Chapter 13

Conclusion

In this thesis we have presented the iAPF, an offline algorithm that approximates an idealized particle filter whose marginal likelihood estimates have zero variance. The main idea is to iteratively approximate a particular sequence of functions, and an empirical study with an implementation using parametric optimization for models with Gaussian transitions showed reasonable performance in some regimes for which the BPF was not able to provide adequate approximations. We applied the iAPF to Bayesian parameter estimation in general state space HMMs by using it as an ingredient in a PMMH Markov chain. It could also conceivably be used in similar, but inexact, noisy Markov chains; Medina-Aguayo et al. (2015) showed that control on the quality of the marginal likelihood estimates can provide theoretical guarantees on the behaviour of the noisy Markov chain. The performance of the iAPF marginal likelihood estimates also suggests they may be useful in simulated maximum likelihood procedures. In our empirical studies, the number of particles used by the iAPF was orders of magnitude smaller than would be required by the BPF for similar approximation accuracy, which may be relevant for models in which space complexity is an issue. In the last part of the thesis we have shown how the iAPF can facilitate statistical inference in the context of diffusion processes, in particular for the problems of estimating transition densities and sampling from diffusion bridges. The modifications to the iAPF we have introduced in Chapter 11 are meant to enhance its performance when a fine Euler–Maruyama discretisation is required. In our simulations the iAPF gives good results even when the BPF performance is compromised due to the degeneracy of the importance weights problem, either because we consider a relatively fine discretisation or because the density point being estimated is relatively far from the mode of the diffusion process transition density. In particular in Section 12.4 we show how in some simple instances the iAPF can be a valid alternative to the MDB approach of Durham & Gallant (2002), as it shows to be more robust with respect to highly non-linear dynamics. An interesting idea to be explored is the possibility to combine the two algorithms: the MDB approach can be used to initialise the iAPF, by producing the first wave of exploring particles and therefore a particles support from which we can shape $\boldsymbol{\psi}_1$. This would mitigate the problem with the degeneracy of the importance weights in the first iterations of the iAPF, possibly more effectively than the complementary iAPF initialisation discussed in Section 11.1.

In the context of likelihood estimation, the perspective brought by viewing the design of particle filters as essentially a function approximation problem has the potential to significantly improve the performance of such methods in a variety of settings. There are, however, a number of alternatives to the parametric optimization approach described in Section 7.3, and it would be of particular future interest to investigate more sophisticated schemes for estimating $\boldsymbol{\psi}^*$, i.e. specific implementations of Algorithm 6.1. We have described a kernel density approach in Section 7.2 and used it to define estimates of the sequence $\boldsymbol{\psi}^*$ with some success in Chapter 8, but the computational cost of the approach was much larger than the parametric approach. Alternatives to the classes $\mathcal{F}$ and Ψ described in Section 6.1 could be obtained using other conjugate families, (see, e.g., Vidoni 1999). We also note that although we restricted the matrix Σ in (7.1) to be diagonal in our examples, the resulting iAPF marginal likelihood estimators performed fairly well in some situations where the optimal sequence $\boldsymbol{\psi}^*$ contained functions that could not be perfectly approximated using any function in the corresponding class. Finally, the stopping rule in the iAPF, described in Algorithm 6.2 and which requires multiple independent marginal likelihood estimates, could be replaced with a stopping rule based on the variance estimators proposed in Lee & Whiteley (2015). For simplicity, we have discussed particle filters in which multinomial resampling is used; a variety of other resampling strategies (see Douc et al. 2005, for a review) can be used instead.

The iAPF performs well in all of our applications and very well in most of them. This is also due to the fact that it effectively exploits the sequential nature of the SSM models, and even if we are dealing with $d \times T$-dimensional distributions π_T which are difficult to approximate, the look-ahead functions are "only" $d-$dimensional and therefore more treatable. It is possible to think of an importance sampling version of the iAPF where an initial proposal density q_0 is used to target an arbitrary density π. The procedure would correspond to the standard version of the iAPF applied to the (degenerate) SSM with initial distribution $\mu = q_0$ and only one potential function $g\left(\cdot\right) = \frac{\pi(\cdot)}{q_0(\cdot)}$. However in applications of interest the target density π tend to be high dimensional - which is a problem for the kernel density estimate approach due to the curse of dimensionality - and multimodal or with a non trivial shape - in which case it might be too much to assume that we can define a class of parametric functions which resemble $g\left(\cdot\right) = \frac{\pi(\cdot)}{q_0(\cdot)}$ with the parametric approach. While some adaptive importance sampling schemes might be more apt in the importance sampling setting (Oh & Berger (1992), Rubinstein (1999)), the iAPF proved to be a competitive new element in the set of sequential Monte Carlo methodologies.

Bibliography

Ait, Y., Kimmel, R. et al. (2007), 'Maximum likelihood estimation of stochastic volatility models', *Journal of Financial Economics* **83**(2), 413–452.

Andrieu, C., Doucet, A. & Holenstein, R. (2010), 'Particle Markov chain Monte Carlo methods', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **72**(3), 269–342.

Andrieu, C. & Roberts, G. O. (2009), 'The Pseudo-Marginal approach for efficient Monte Carlo computations', *The Annals of Statistics* pp. 697–725.

Andrieu, C. & Vihola, M. (2015), 'Convergence properties of pseudo-marginal Markov chain Monte Carlo algorithms', *The Annals of Applied Probability* **25**(2), 1030–1077.

Ascher, U. M. & Petzold, L. R. (1998), *Computer methods for ordinary differential equations and differential-algebraic equations*, Society for Industrial and Applied Mathematics, Philadelphia. Literaturverz. S. 299 - 305.

Beaumont, M. A. (2003), 'Estimation of population growth or decline in genetically monitored populations', *Genetics* **164**(3), 1139–1160.

Bérard, J., Del Moral, P. & Doucet, A. (2014), 'A lognormal central limit theorem for particle approximations of normalizing constants', *Electronic Journal of Probability* **19**(94), 1–28.

Beskos, A. & Roberts, G. O. (2005), 'Exact simulation of diffusions', *The Annals of Applied Probability* .

Black, F. & Scholes, M. (1973), 'The pricing of options and corporate liabilities', *The journal of political economy* pp. 637–654.

Bromiley, P. (2003), ‘Products and convolutions of gaussian probability density functions’, *Tina-Vision Memo* **3**(4).

Cappé, O., Godsill, S. J. & Moulines, E. (2007), ‘An overview of existing methods and recent advances in sequential monte carlo’, *Proceedings of the IEEE* **95**(5), 899–924.

Carpenter, J., Clifford, P. & Fearnhead, P. (1999), ‘Improved particle filter for nonlinear problems’, *IEE Proceedings-Radar, Sonar and Navigation* **146**(1), 2–7.

Chan, K. C., Karolyi, G. A., Longstaff, F. A. & Sanders, A. B. (1992), ‘An empirical comparison of alternative models of the short-term interest rate’, *The Journal of Finance* **47**(3), 1209–1227.

Chen, R., Wang, X. & Liu, J. S. (2000), ‘Adaptive joint detection and decoding in flat-fading channels via mixture Kalman filtering’, *Information Theory, IEEE Transactions on* **46**(6), 2079–2094.

Chib, S., Omori, Y. & Asai, M. (2009), Multivariate stochastic volatility, *in* T. G. Andersen, R. A. Davis, J.-P. Kreiss & T. V. Mikosch, eds, ‘Handbook of Financial Time Series’, Springer, pp. 365–400.

Chiu, S.-T. (1991), ‘Bandwidth selection for kernel density estimation’, *The Annals of Statistics* pp. 1883–1905.

Chopin, N. (2004), ‘Central limit theorem for sequential Monte Carlo methods and its application to Bayesian inference’, *The Annals of Statistics* pp. 2385–2411.

Chopin, N. (2007), ‘Inference and model choice for sequentially ordered hidden markov models’, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **69**(2), 269–284.

Clapp, T. C. & Godsill, S. J. (1999), ‘Fixed-lag smoothing using sequential importance sampling’, *Bayesian Statistics 6: Proceedings of the Sixth Valencia International Meeting* **6**, 743–752.

Coffey, W., Kalmykov, Y. P. & Waldron, J. (2004), 'The langevin equation: with applications to stochastic problems in physics', *Chemistry and Electrical Engineering. World Scientific* .

Cox, J. C., Ingersoll Jr, J. E. & Ross, S. A. (1985), 'A theory of the term structure of interest rates', *Econometrica: Journal of the Econometric Society* pp. 385–407.

Del Moral, P. (2004), *Feynman-Kac Formulae*, Springer.

Del Moral, P., Doucet, A. & Jasra, A. (2006), 'Sequential Monte Carlo samplers', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **68**(3), 411–436.

Del Moral, P. & Guionnet, A. (1999), 'Central limit theorem for nonlinear filtering and interacting particle systems', *The Annals of Applied Probability* **9**(2), 275–297.

Del Moral, P., Jacod, J. & Protter, P. (2001), 'The Monte-Carlo method for filtering with discrete-time observations', *Probability Theory and Related Fields* **120**(3), 346.
URL: *http://dx.doi.org/10.1007/PL00008786*

Diggle, P. J. & Gratton, R. J. (1984), 'Monte Carlo methods of inference for implicit statistical models', *Journal of the Royal Statistical Society. Series B (Methodological)* pp. 193–227.

Douc, R., Cappé, O. & Moulines, E. (2005), Comparison of resampling schemes for particle filtering, *in* 'Image and Signal Processing and Analysis, 2005. ISPA 2005. Proceedings of the 4th International Symposium on', IEEE, pp. 64–69.

Douc, R. & Moulines, E. (2008), 'Limit theorems for weighted samples with applications to sequential Monte Carlo methods', *The Annals of Statistics* **36**(5), 2344–2376.

Doucet, A., Briers, M. & Sénécal, S. (2006), 'Efficient block sampling strategies for sequential Monte Carlo methods', *Journal of Computational and Graphical Statistics* **15**(3).

Doucet, A., De Freitas, N. & Gordon, N. (2001), An introduction to sequential monte carlo methods, *in* 'Sequential Monte Carlo methods in practice', Springer, pp. 3–14.

Doucet, A., Godsill, S. & Andrieu, C. (2000), 'On sequential Monte Carlo sampling methods for Bayesian filtering', *Statistics and Computing* **10**(3), 197–208.

Doucet, A. & Johansen, A. M. (2011), A tutorial on particle filtering and smoothing: Fiteen years later, *in* D. Crisan & B. Rozovsky, eds, 'The Oxford Handbook of Nonlinear Filtering', pp. 656–704.

Doucet, A., Pitt, M., Deligiannidis, G. & Kohn, R. (2015), 'Efficient implementation of Markov chain Monte Carlo when using an unbiased likelihood estimator', *Biometrika* **102**(2), 295–313.

Durham, G. B. & Gallant, A. R. (2002), 'Numerical techniques for maximum likelihood estimation of continuous-time diffusion processes', *Journal of Business & Economic Statistics* **20**(3), 297–338.

Fearnhead, P. (2004), 'Particle filters for mixture models with an unknown number of components', *Statistics and Computing* **14**(1), 11–21.

Fearnhead, P. (2008), 'Computational methods for complex stochastic systems: a review of some alternatives to mcmc', *Statistics and Computing* **18**(2), 151–171.

Fernández-Villaverde, J. & Rubio-Ramírez, J. F. (2007), 'Estimating macroeconomic models: A likelihood approach', *The Review of Economic Studies* **74**(4), 1059–1087.

Gamerman, D. & Lopes, H. (2006), 'Monte carlo markov chain: stochastic simulation for bayesian inference', *Monte Carlo Markov Chain: stochastic simulation for bayesian inference* .

Ghahramani, Z. (1998), Learning dynamic Bayesian networks, *in* 'Adaptive processing of sequences and data structures', Springer, pp. 168–197.
URL: *http://link.springer.com/chapter/10.1007/BFb0053999*

Gikhman, I. I. & Skorokhod, A. V. (1969), *Introduction to the theory of random processes*, Saunders mathematical books, Saunders.

Godsill, S. & Vermaak, J. (2004), Models and algorithms for tracking using trans-dimensional sequential monte carlo, *in* 'Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04). IEEE International Conference on', Vol. 3, IEEE, pp. iii–976.

Golightly, A. & Wilkinson, D. J. (2011), 'Bayesian parameter inference for stochastic biochemical network models using particle Markov chain Monte Carlo', *Interface focus* p. rsfs20110047.

Gordon, N. J., Salmond, D. J. & Smith, A. F. (1993), 'Novel approach to nonlinear/non-Gaussian Bayesian state estimation', *IEE Proceedings-Radar, Sonar and Navigation* **140**(2), 107–113.

Green, P. J. (1995), 'Reversible jump markov chain monte carlo computation and bayesian model determination', *Biometrika* pp. 711–732.

Guarniero, P., Johansen, A. M. & Lee, A. (2016), 'The iterated auxiliary particle filter', *Journal of the American Statistical Association* (just-accepted).

Harvey, A., Ruiz, E. & Shephard, N. (1994), 'Multivariate stochastic variance models', *The Review of Economic Studies* **61**(2), 247–264.

Hastings, W. K. (1970), 'Monte carlo sampling methods using markov chains and their applications', *Biometrika* **57**(1), 97–109.

Heston, S. L. (1993), 'A closed-form solution for options with stochastic volatility with applications to bond and currency options', *Review of financial studies* **6**(2), 327–343.

Hürzeler, M. & Künsch, H. R. (2001), Approximating and maximising the likelihood for a general state-space model, *in* A. Doucet, N. de Freitas & N. Gordon, eds, 'Sequential Monte Carlo methods in practice', Springer, pp. 159–175.

Ikeda, N. & Watanabe, S. (1981), *Stochastic differential equations and diffusion processes*, number 24 *in* '@North-Holland mathematical library', North-Holland Publ. [u.a.], Amsterdam [u.a.]. Literaturverz. S. 453 - 460.

Johansen, A. M. & Doucet, A. (2008), 'A note on auxiliary particle filters', *Statistics & Probability Letters* **78**(12), 1498–1504.

Kappen, H. J., Gómez, V. & Opper, M. (2012), 'Optimal control as a graphical model inference problem', *Machine learning* **87**(2), 159–182.

Kim, S., Shephard, N. & Chib, S. (1998), 'Stochastic volatility: likelihood inference and comparison with arch models', *The Review of Economic Studies* **65**(3), 361–393.

Kitagawa, G. (1998), 'A self-organizing state-space model', *Journal of the American Statistical Association* pp. 1203–1215.

Kloeden, P. E. & Platen, E. (1992), *Numerical solution of stochastic differential equations*, number 23 *in* 'Applications of mathematics', Springer, Berlin [u.a.]. Literaturverz. S. [597] - 624.

Kong, A., Liu, J. S. & Wong, W. H. (1994), 'Sequential imputations and Bayesian missing data problems', *Journal of the American Statistical Association* **89**(425), 278–288.

Künsch, H. (2005), 'Recursive Monte Carlo filters: algorithms and theoretical analysis', *The Annals of Statistics* **33**(5), 1983–2021.

Künsch, H. R. (2000), State space and hidden markov models, *in* 'Complex stochastic systems', Chapman and Hall/CRC.

Lee, A. & Łatuszyński, K. (2014), 'Variance bounding and geometric ergodicity of Markov chain Monte Carlo kernels for approximate Bayesian computation.', *Biometrika* **101**(3), 655–671.

Lee, A. & Whiteley, N. (2015), 'Variance estimation and allocation in the particle filter', *arXiv preprint arXiv:1509.00394* .

Lerman, S. & Manski, C. (1981), On the use of simulated frequencies to approximate choice probabilities, *in* 'Structural analysis of discrete data with econometric applications', The MIT press, pp. 305–319.

Lin, M., Chen, R., Liu, J. S. et al. (2013), 'Lookahead strategies for sequential Monte Carlo', *Statistical Science* **28**(1), 69–94.

Liu, J. S. & Chen, R. (1995), 'Blind deconvolution via sequential imputations', *Journal of the American Statistical Association* **90**, 567–576.

Liu, J. & West, M. (2001), Combined parameter and state estimation in simulation-based filtering, *in* A. Doucet, N. de Freitas & N. Gordon, eds, 'Sequential Monte Carlo methods in practice', Springer, pp. 197–223.

Loader, C. R. (1999), 'Bandwidth selection: classical or plug-in?', *Annals of Statistics* pp. 415–438.

Malory, S. & Sherlock, C. (2016), 'Residual-bridge constructs for conditioned diffusions', *arXiv preprint arXiv:1602.04439* .

McAdams, H. H., Arkin, A. P. & Shapiro, L. (1999), 'System and method for simulating operation of biochemical systems'. US Patent 5,914,891.

Medina-Aguayo, F. J., Lee, A. & Roberts, G. O. (2015), 'Stability of noisy Metropolis–Hastings', *arXiv preprint arXiv:1503.07066* .

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. & Teller, E. (1953), 'Equation of state calculations by fast computing machines', *The journal of chemical physics* **21**(6), 1087–1092.

Nadaraya, E. A. (1964), 'On estimating regression', *Theory of Probability & Its Applications* **9**(1), 141–142.

Oh, M.-S. & Berger, J. O. (1992), 'Adaptive importance sampling in monte carlo integration', *Journal of Statistical Computation and Simulation* **41**(3-4), 143–168.

Parzen, E. (1962), 'On estimation of a probability density function and mode', *The Annals of Mathematical Statistics* pp. 1065–1076.

Pedersen, A. R. (1995), ‘A new approach to maximum likelihood estimation for stochastic differential equations based on discrete observations’, *Scandinavian journal of statistics* pp. 55–71.

Pitt, M. K. & Shephard, N. (1999), ‘Filtering via simulation: Auxiliary particle filters’, *Journal of the American Statistical Association* **94**(446), 590–599.

Revuz, D. & Yor, M. (1994), *Continuous martingales and Brownian motion*, Series of comprehensive studies in mathematics, 2nd edn, Springer.

Richard, J.-F. & Zhang, W. (2007), ‘Efficient high-dimensional importance sampling’, *Journal of Econometrics* **141**(2), 1385–1411.

Robert, C. P. & Casella, G. (2005), ‘Monte carlo statistical methods’.

Roberts, G. O. & Tweedie, R. L. (1996), ‘Geometric convergence and central limit theorems for multidimensional hastings and metropolis algorithms’, *Biometrika* pp. 95–110.

Rosenblatt, M. et al. (1956), ‘Remarks on some nonparametric estimates of a density function’, *The Annals of Mathematical Statistics* **27**(3), 832–837.

Rubinstein, R. (1999), ‘The cross-entropy method for combinatorial and continuous optimization’, *Methodology and computing in applied probability* **1**(2), 127–190.

Ruiz, H.-C. & Kappen, H. J. (2017), ‘Particle smoothing for hidden diffusion processes: Adaptive path integral smoother’, *IEEE Transactions on Signal Processing* .

Scharth, M. & Kohn, R. (2016), ‘Particle efficient importance sampling’, *Journal of Econometrics* **190**(1), 133–147.

Sherlock, C., Thiery, A. H., Roberts, G. O. & Rosenthal, J. S. (2015), ‘On the efficiency of pseudo-marginal random walk Metropolis algorithms’, *The Annals of Statistics* **43**(1), 238–275.

Silverman, B. (1982), ‘Algorithm as 176: Kernel density estimation using the fast fourier transform’, *Applied Statistics* pp. 93–99.

Tikhonov, A. N., Goncharsky, A. V., Stepanov, V. V. & Yagola, A. G. (1995), 'Regularization methods', *Numerical Methods for the Solution of Ill-Posed Problems* .

Tran, M.-N., Scharth, M., Pitt, M. K. & Kohn, R. (2014), 'Importance sampling squared for bayesian inference in latent variable models'.

Vidoni, P. (1999), 'Exponential family state space models based on a conjugate latent process', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **61**(1), 213–221.

Walker, A. J. (1974), 'New fast method for generating discrete random numbers with arbitrary frequency distributions', *Electronics Letters* **10**(8), 127–128.

Walker, A. J. (1977), 'An efficient method for generating discrete random variables with general distributions', *ACM Transactions on Mathematical Software* **3**(3), 253–256.

Wang, B. & Wang, X. (2007), 'Bandwidth selection for weighted kernel density estimation', *arXiv preprint arXiv:0709.1616* .

Watson, G. S. (1964), 'Smooth regression analysis', *Sankhyā: The Indian Journal of Statistics, Series A* **26**(4), 359–372.

Whitaker, G. A., Golightly, A., Boys, R. J. & Sherlock, C. (2016), 'Improved bridge constructs for stochastic differential equations', *Statistics and Computing* pp. 1–16.

Zhang, J. L. & Liu, J. S. (2002), 'A new sequential importance sampling method and its application to the two-dimensional hydrophobic–hydrophilic model', *The Journal of Chemical Physics* **117**(7), 3492–3498.