Forward Smoothing using Sequential Monte Carlo

Pierre Del Moral*, Arnaud Doucet†, Sumeetpal S. Singh‡

Technical Report CUED/F-INFENG/TR 638, Cambridge University.
First version September 27th 2009, Second version November 24th 2009, Third version December 12th 2010.

Abstract

Sequential Monte Carlo (SMC) methods are a widely used set of computational tools for inference in non-linear non-Gaussian state-space models. We propose a new SMC algorithm to compute the expectation of additive functionals recursively. Essentially, it is an online or “forward-only” implementation of a forward filtering backward smoothing SMC algorithm proposed in [18]. Compared to the standard path space SMC estimator whose asymptotic variance increases quadratically with time even under favourable mixing assumptions, the asymptotic variance of the proposed SMC estimator only increases linearly with time. This forward smoothing procedure allows us to implement on-line maximum likelihood parameter estimation algorithms which do not suffer from the particle path degeneracy problem.

Some key words: Expectation-Maximization, Forward Filtering Backward Smoothing, Recursive Maximum Likelihood, Sequential Monte Carlo, Smoothing, State-Space Models.

1 Introduction

1.1 State-space models and inference aims

State-space models (SSM) are a very popular class of non-linear and non-Gaussian time series models in statistics, econometrics and information engineering; see for example [7], [19], [20]. An SSM is comprised of a pair of discrete-time stochastic processes, \( \{X_n\}_{n \geq 0} \) and \( \{Y_n\}_{n \geq 0} \), where the former is an \( X \)-valued unobserved process and the latter is a \( Y \)-valued process which is observed. The hidden process \( \{X_n\}_{n \geq 0} \) is a Markov process with initial density \( \mu_{\theta}(x) \) and Markov transition density \( f_{\theta}(x' | x) \), i.e.

\[
X_0 \sim \mu_{\theta}(\cdot) \quad \text{and} \quad X_n | (X_{n-1} = x_{n-1}) \sim f_{\theta}(\cdot | x_{n-1}), \quad n \geq 1.
\] (1.1)

*Centre INRIA Bordeaux et Sud-Ouest & Institut de Mathématiques de Bordeaux , Université de Bordeaux I, 351 cours de la Libération 33405 Talence cedex, France (Pierre.Del-Moral@inria.fr)
†The Institute of Statistical Mathematics, 4-6-7 Minami-Azabu, Minato-ku, Tokyo 106-8569, Japan and Department of Statistics & Department of Computer Science, University of British Columbia, 333-6356 Agricultural Road, Vancouver, BC, V6T 1Z2, Canada (Arnaud@stat.ubc.ca)
‡Department of Engineering, University of Cambridge, Trumpington Street, CB2 1PZ, United Kingdom (sss40@cam.ac.uk)
It is assumed that the observations \( \{Y_n\}_{n \geq 0} \) conditioned upon \( \{X_n\}_{n \geq 0} \) are statistically independent and have marginal density \( g_\theta(y|x) \), i.e.

\[
Y_n | \{X_k\}_{k \geq 0} = \{x_k\}_{k \geq 0} \sim g_\theta(\cdot|x_n). \tag{1.2}
\]

We also assume that \( \mu_\theta(x) \), \( f_\theta(x|x') \) and \( g_\theta(y|x) \) are densities with respect to (w.r.t.) suitable dominating measures denoted generically as \( dx \) and \( dy \). For example, if \( X \subseteq \mathbb{R}^p \) and \( Y \subseteq \mathbb{R}^q \) then the dominating measures could be the Lebesgue measures. The variable \( \theta \) in the densities of these random variables are the particular parameters of the model. The set of possible values for \( \theta \) is denoted \( \Theta \). The model (1.1)-(1.2) is also referred to as a Hidden Markov Model (HMM) in the literature.

For any sequence \( \{z_n\}_{n \in \mathbb{Z}} \) and integers \( j \geq i \), let \( z_{i:j} \) denote the set \( \{z_i, z_{i+1}, \ldots, z_j\} \). (When \( j < i \) this is to be understood as the empty set.) Equations (1.1) and (1.2) define the joint density of \( \{X_{0:n}, Y_{0:n}\} \),

\[
p_\theta(x_{0:n}, y_{0:n}) = \mu_\theta(x_0) \prod_{k=1}^{n} f_\theta(x_k | x_{k-1}) \prod_{k=0}^{n} g_\theta(y_k | x_k), \tag{1.3}
\]

which yields the marginal likelihood,

\[
p_\theta(y_{0:n}) = \int p_\theta(x_{0:n}, y_{0:n}) \, dx_{0:n}. \tag{1.4}
\]

Let \( s_k : X \times X \to \mathbb{R}, k \in \mathbb{N} \), be a sequence of functions and \( S_n : X^n \to \mathbb{R}, n \in \mathbb{N} \), be the corresponding sequence of additive functionals constructed from \( s_k \) as follows:

\[
S_n(x_{0:n}) = \sum_{k=1}^{n} s_k(x_{k-1}, x_k). \tag{1.5}
\]

There are many instances where it is necessary to be able to compute the following expectations recursively in time,

\[
\mathcal{S}_{\theta}^n = \mathbb{E}_\theta [S_n(X_{0:n}) | y_{0:n}]. \tag{1.6}
\]

The conditioning implies the expectation should be computed w.r.t. the density of \( X_{0:n} \) given \( Y_{0:n} = y_{0:n} \), i.e. \( p_\theta(x_{0:n}, y_{0:n}) \propto p_\theta(x_{0:n}, y_{0:n}) \) and for this reason \( \mathcal{S}_{\theta}^n \) is referred to as a smoothed additive functional.

As the first example of the need to perform such computations, consider the problem of computing the score vector, \( \nabla \log p_\theta(y_{0:n}) \). The score is a vector in \( \mathbb{R}^d \) and its \( i \)th component is

\[
[\nabla \log p_\theta(y_{0:n})]_i = \frac{\partial \log p_\theta(y_{0:n})}{\partial \theta^i}. \tag{1.7}
\]

Using Fisher’s identity, the problem of computing the score becomes an instance of (1.6), i.e.

\[
\nabla \log p_\theta(y_{0:n}) = \sum_{k=1}^{n} \mathbb{E}_\theta [\nabla \log f_\theta(X_k | X_{k-1}) | y_{0:n}] + \sum_{k=0}^{n} \mathbb{E}_\theta [\nabla \log g_\theta(y_k | X_k) | y_{0:n}] + \mathbb{E}_\theta [\nabla \log \mu_\theta(X_0) | y_{0:n}] \tag{1.8}
\]

\footnote{Incorporating dependency of \( s_k \) on \( y_k \), i.e. \( S_n \) is the sums of term of the form \( s_k(x_{k-1}, x_k, y_k) \), is merely a matter of redefining \( s_k \) in the computations to follow.}
An alternative representation of the score as a smoothed additive functional based on infinitesimal perturbation analysis is given in [12]. The score has applications to Maximum Likelihood (ML) parameter estimation [32], [37].

The second example is ML parameter estimation using the Expectation-Maximization (EM) algorithm. Let \( y_{0:n} \) be a batch of data and the aim is to maximise \( p_\theta (y_{0:n}) \) w.r.t. \( \theta \). Given a current estimate \( \theta' \), a new estimate \( \theta'' \) is obtained by maximizing the function

\[
Q(\theta', \theta) = \sum_{k=1}^n \mathbb{E}_{\theta'}[\log f_\theta (X_k | X_{k-1}) | y_{0:n}] + \sum_{k=0}^n \mathbb{E}_{\theta'}[\log g_\theta (y_k | X_k) | y_{0:n}]
+ \mathbb{E}_{\theta'}[\log \mu_\theta (X_0) | y_{0:n}]
\]

w.r.t. \( \theta \) and setting \( \theta'' \) to the maximising argument. A fundamental property of the EM algorithm is \( p_{\theta''} (y_{0:n}) \geq p_{\theta'} (y_{0:n}) \). For linear Gaussian models and finite state-space HMM, it is possible to perform the computations in the definition of \( Q(\theta', \theta) \). For general non-linear non-Gaussian state-space models of the form (1.1)-(1.2), we need to rely on numerical approximation schemes.

### 1.2 Current approaches to smoothing with SMC

SMC methods are a class of algorithms that sequentially approximate the sequence of posterior distributions \( \{p_\theta (dx_{0:n} | y_{0:n})\}_{n \geq 0} \) using a set of \( N \) weighted random samples called particles. Specifically, the SMC approximation of \( p_\theta (dx_{0:n} | y_{0:n}) \), for \( n \geq 0 \), is

\[
\hat{p}_\theta (dx_{0:n} | y_{0:n}) := \sum_{i=1}^N W_n^{(i)} \delta_{X_{0:n}^{(i)}} (dx_{0:n}), \quad W_n^{(i)} \geq 0, \quad \sum_{i=1}^N W_n^{(i)} = 1, \quad (1.9)
\]

where \( W_n^{(i)} \) is the importance weight associated to particle \( X_{0:n}^{(i)} \) and \( \delta_{X_{0:n}^{(i)}} \) is the Dirac measure with an atom at \( X_{0:n}^{(i)} \). The particles are propagated forward in time using a combination of importance sampling and resampling steps and there are several variants of both these steps; see [14], [19] for details. SMC methods are parallelisable and flexible, the latter in the sense that SMC approximations of the posterior densities for a variety of SSMs can be constructed quite easily. SMC methods were popularized by the many successful applications to SSM.

#### 1.2.1 Path space and fixed-lag approximations

A SMC approximation of \( S_0^n \) may be constructed by replacing \( p_\theta (dx_{0:n} | y_{0:n}) \) in Eq. (1.6) with its SMC approximation in Eq. (1.9) - we call this the path space method since the SMC approximation of \( p_\theta (dx_{0:n} | y_{0:n}) \), which is a probability distribution on \( \mathcal{X}_{n+1} \), is used. Fortunately there is no need to store the entire ancestry of each particle, i.e. \( \{X_{0:n}^{(i)}\}_{1 \leq i \leq N} \), which would require a growing memory. Also, this estimate can be computed recursively. However, the reliance on the approximation of the joint distribution \( p_\theta (dx_{0:n} | y_{0:n}) \) is bad. It is well-known in the SMC literature that the approximation of \( p_\theta (dx_{0:n} | y_{0:n}) \) becomes progressively impoverished as \( n \) increases because of the successive resampling steps [8], [13], [34]. That is, the number of distinct samples representing \( p_\theta (dx_{0:k} | y_{0:n}) \) for any fixed \( k < n \)
diminishes as \( n \) increases – this is known as the particle path degeneracy problem. Hence, whatever being the number of particles, \( p_\theta (dx_{0:k} \mid y_{0:n}) \) will eventually be approximated by a single unique particle for all (sufficiently large) \( n \). This has severe consequences for the SMC estimate \( S^\theta_n \). In \cite{13}, under favourable mixing assumptions, the authors established an upper bound on the \( L^p \) error which is proportional to \( n^2/\sqrt{N} \). Under similar assumptions, it was shown in \cite{37} that the asymptotic variance of this estimate increases at least quadratically with \( n \). To reduce the variance, alternative methods have been proposed. The technique proposed in \cite{29} relies on the fact that for a SSM with “good” forgetting properties,

\[
p_\theta (x_{0:k} \mid y_{0:n}) \approx p_\theta (x_{0:k} \mid y_{0:\min(k+\Delta,n)})
\]

when the horizon \( \Delta \) is large enough; that is observations collected after times \( k+\Delta \) bring little additional information about \( X_{0:k} \). (See \cite{16, Corollary 2.9} for exponential error bounds.) This suggests that a very simple scheme to curb particle degeneracy is to stop updating the SMC estimate beyond time \( k + \Delta \). This algorithm is trivial to implement but the main practical problem is that of determining an appropriate value for \( \Delta \) such that the two densities in Eq. (1.10) are close enough and particle degeneracy is low. These are conflicting requirements. A too small value for the horizon will result in \( p_\theta (x_{0:k} \mid y_{0:\min(k+\Delta,n)}) \) being a poor approximation of \( p_\theta (x_{0:k} \mid y_{0:n}) \) but the particle degeneracy will be low. On the other hand, a larger horizon improves the approximation in Eq. (1.10) but particle degeneracy will creep in. Automating the selection of \( \Delta \) is difficult. Additionally, for any finite \( \Delta \) the SMC estimate of \( S^\theta_n \) will suffer from a non vanishing bias even as \( N \to \infty \). In \cite{34}, for an optimized value of \( \Delta \) which is dependent on \( n \) and the typically unknown mixing properties of the model, the SMC estimates of \( S^\theta_n \) based on the approximation in Eq. (1.10) were shown to have an \( L^p \) error and bias upper bounded by quantities proportional to \( n \log n/\sqrt{N} \) and \( n \log n/N \) under regularity assumptions.

The computational cost of the SMC approximation of \( S^\theta_n \) computed using either the path space method or the truncated horizon method of \cite{29} is \( \mathcal{O}(N) \).

### 1.2.2 Approximating the smoothing equations

A standard alternative to computing \( S^\theta_n \) is to use SMC approximations of fixed-interval smoothing techniques such as the Forward Filtering Backward Smoothing (FFBS) algorithm \cite{18, 26}. Theoretical results on the SMC approximations of the FFBS algorithm have been recently established in \cite{17}; this includes a central limit theorem and exponential deviation inequalities. In particular, under appropriate mixing assumptions, the authors have obtained time-uniform deviation inequalities for the SMC-FFBS approximations of the marginals \( \{p_\theta (dx_k \mid y_{0:n})\}_{0 \leq k \leq n} \) \cite[Section 5]{17}; see \cite{15} for alternative proofs and complementary results. Let \( \hat{S}^\theta_n \) denote the SMC-FFBS estimate of \( S^\theta_n \). In this work it is established that the asymptotic variance of \( \sqrt{N} (\hat{S}^\theta_n - S^\theta_n) \) only grows linearly with time \( n \); a fact which was also alluded to in \cite{15}. The main advantage of the SMC implementation of the FFBS algorithm is that it does not have any tuning parameter other than the number of particles \( N \). However, the improved theoretical properties comes at a computational price; this algorithm has a computational complexity of \( \mathcal{O}(N^2) \) compared to \( \mathcal{O}(N) \) for the methods previously discussed. (It is possible to use fast computational methods to reduce the computational cost to \( \mathcal{O}(N \log N) \) \cite{30}.) Another restriction is that the SMC implementation of the FFBS algorithm does not yield an online algorithm.
1.3 Contributions and organization of the article

The contributions of this article are as follows.

- We propose an original online implementation of the SMC-FFB S estimate of $S^\theta_n$. A particular case of this new algorithm was presented in [36], [37] to compute the score vector (1.7). However, because it was catered to estimating the score, the authors failed to realise its full generality.

- An upper bound for the non-asymptotic mean square error of the SMC-FFBS estimate $\hat{S}_n^\theta$ of $S_n^\theta$ is derived under regularity assumptions. It follows from this bound that the asymptotic variance of $\sqrt{N}(\hat{S}_n^\theta - S_n^\theta)$ is bounded by a quantity proportional to $n$. This complements results recently obtained in [15], [17].

- We demonstrate how the online implementation of the SMC-FFB S estimate of $S_n^\theta$ can be applied to the problem of recursively estimating the parameters of a SSM from data. We present original SMC implementations of Recursive Maximum Likelihood (RML) [32], [36], [39] and of the on-line EM algorithm [25], [22], [23], [33], [8], [28, Section 3.2]. These SMC implementations do not suffer from the particle path degeneracy problem.

The remainder of this paper is organized as follows. In Section 2 the standard FFBS recursion and its SMC implementation is presented. It is then shown how this recursion and its SMC implementation can be implemented exactly with only a forward pass. A non-asymptotic variance bound is presented in Section 3. Recursive parameter estimation procedures are presented in Section 4 and numerical results are given in Section 5. We conclude in Section 6 and the proof of the main theoretical result is given in the Appendix.

2 Forward smoothing and SMC approximations

We first review the standard FFBS recursion and its SMC approximation [18], [26]. This is then followed by a derivation of a forward-only version of the FFBS recursion and its corresponding SMC implementation. The algorithms presented in this section do not depend on any specific SMC implementation to approximate $\{p_\theta(dx_n|y_{0:n})\}_{n\geq 0}$.

2.1 The forward filtering backward smoothing recursion

Recall the definition of $S^\theta_n$ in Eq. (1.6). The standard FFBS procedure to compute $S^\theta_n$ proceeds in two steps. In the first step, which is the forward pass, the filtering densities $\{p_\theta(x_k|y_{0:k})\}_{0\leq k\leq n}$ are computed using Bayes’ formula:

$$
p_\theta(x_{k+1}|y_{0:k+1}) = \frac{g_\theta(y_{k+1}|x_{k+1}) \int \int \int \int f_\theta(x_{k+1}|x_k) p_\theta(x_k|y_{0:k}) dx_k}{\int \int \int \int \int g_\theta(y_{k+1}|x_{k+1}) f_\theta(x_{k+1}|x_k) p_\theta(x_k|y_{0:k}) dx_k dx_{k+1}}.
$$

The second step is the backward pass that computes the following marginal smoothed densities which are needed to evaluate each term in the sum that defines $S^\theta_n$:

$$
p_\theta(x_{k-1}, x_k|y_{0:n}) = p_\theta(x_k|y_{0:n}) p_\theta(x_{k-1}|y_{0:k-1}, x_k), \quad 1 \leq k \leq n. \tag{2.1}
$$
where
\[
p_{\theta}(x_{k-1} \mid y_{0:k-1}, x_k) = \frac{f_{\theta}(x_k \mid x_{k-1}) p_{\theta}(x_{k-1} \mid y_{0:k-1})}{p_{\theta}(x_k \mid y_{0:k-1})}.
\]

We compute Eq. (2.1) commencing at \( k = n \) and then, decrementing \( k \) each time, until \( k = 1 \). (Integrating Eq. (2.1) w.r.t. \( x_k \) will yield \( p_{\theta}(x_{k-1} \mid y_{0:n}) \) which is needed for the next computation.) To compute \( S_n^{\theta} \), \( n \) backward steps must be executed and then \( n \) expectations computed. This must then be repeated at time \( n+1 \) to incorporate the effect of the new observation \( y_{n+1} \) on these calculations. Clearly this is not an online procedure for computing \( \{S_n^{\theta}\}_{n \geq 1} \).

The SMC implementation of the FFBS recursion is straightforward [18]. In the forward pass, we compute and store the SMC approximation \( \hat{p}_{\theta}(dx_k \mid y_{0:k}) \) of \( p_{\theta}(dx_k \mid y_{0:k}) \) for \( k = 0, 1, \ldots, n \). In the backward pass, we simply substitute this SMC approximation in the place of \( p_{\theta}(dx_k \mid y_{0:k}) \) in Eq. (2.1). Let
\[
\hat{p}_{\theta}(dx_k \mid y_{0:n}) = \sum_{i=1}^{N} W_{k}^{(i)} \delta_{X_k^{(i)}}(dx_k)
\]
be the SMC approximation of \( p_{\theta}(dx_k \mid y_{0:n}), k \leq n, \) initialised at \( k = n \) by setting \( W_{n}^{(i)} = W_{n}^{(i)} \). By substituting \( \hat{p}_{\theta}(dx_{k-1} \mid y_{0:k-1}) \) for \( p_{\theta}(x_{k-1} \mid y_{0:k-1}) \) in Eq. (2.2), we obtain
\[
\hat{p}_{\theta}(dx_{k-1} \mid y_{0:k-1}, x_k) = \frac{\sum_{i=1}^{N} W_{k-1}^{(i)} f_{\theta}(x_k \mid X_{k-1}^{(i)}) \delta_{X_k^{(i)}}(dx_k)}{\sum_{i=1}^{N} W_{k-1}^{(i)} f_{\theta}(x_k \mid X_{k-1}^{(i)})}.
\]

This approximation is combined with \( \hat{p}_{\theta}(dx_k \mid y_{0:n}) \) (see Eq. (2.1)) to obtain
\[
\hat{p}_{\theta}(dx_{k-1:k} \mid y_{0:n}) = \sum_{i=1}^{N} \sum_{j=1}^{N} W_{k-1}^{(i)} W_{k}^{(j)} \frac{f_{\theta}(X_k^{(j)} \mid X_{k-1}^{(i)})}{f_{\theta}(X_k^{(j)} \mid X_{k-1}^{(i)})} \delta_{X_{k-1}^{(i)}, X_k^{(j)}}(dx_{k-1:k}).
\]
Marginalising this approximation will give the approximation to \( \hat{p}_{\theta}(dx_{k-1} \mid y_{0:n}) \), that is \( \{W_{k-1}^{(i)}, X_{k-1}^{(i)}\}_{1 \leq i \leq N} \), where
\[
W_{k-1}^{(i)} = \sum_{j=1}^{N} W_{k}^{(j)} \frac{W_{k}^{(j)} f_{\theta}(X_k^{(j)} \mid X_{k-1}^{(i)})}{f_{\theta}(X_k^{(j)} \mid X_{k-1}^{(i)})}.
\]
The SMC estimate \( \hat{S}_n^{\theta} \) of \( S_n^{\theta} \) is then given by
\[
\hat{S}_n^{\theta} = \sum_{k=1}^{n} \int s_k(x_{k-1}, x_k) \hat{p}_{\theta}(dx_{k-1:k} \mid y_{0:n}).
\]
The backward recursion for the weights, given in Eq. (2.6), makes this an off-line algorithm for computing \( \hat{S}_n^{\theta} \).
2.2 A forward only version of the forward filtering backward smoothing recursion

To circumvent the need for the backward pass in the computation of $S_n^\theta$, the following auxiliary function (on $X$) is introduced,

$$ T_n^\theta (x_n) := \int S_n (x_{0:n}) \ p_\theta (x_{0:n-1} \mid y_{0:n-1}, x_n) \ dx_{0:n-1}. \quad (2.8) $$

It is apparent that

$$ S_n^\theta = \int T_n^\theta (x_n) \ p_\theta (x_n \mid y_{0:n}) \ dx_n. \quad (2.9) $$

The following proposition establishes a forward recursion to compute $\{ T_n^\theta \}_{n \geq 0}$, which is henceforth referred to as the forward smoothing recursion. For sake of completeness, the proof of this proposition is given.

**Proposition 2.1** For $n \geq 1$, we have

$$ T_n^\theta (x_n) = \int \left[ T_{n-1}^\theta (x_{n-1}) + s_n (x_{n-1}, x_n) \right] p_\theta (x_{n-1} \mid y_{0:n-1}, x_n) \ dx_{n-1}, \quad (2.10) $$

where $T_0^\theta (x_0) := 0$.

**Proof.** The proof is straightforward

$$ T_n^\theta (x_n) := \int [S_{n-1} (x_{0:n-1}) + s_n (x_{n-1}, x_n)] \ p_\theta (x_{0:n-1} \mid y_{0:n-1}, x_n) \ dx_{0:n-1} $$

$$ = \int \left[ \int S_{n-1} (x_{0:n-1}) p_\theta (x_{0:n-2} \mid y_{0:n-2}, x_{n-1}) \ dx_{0:n-2} \right] p_\theta (x_{n-1} \mid y_{0:n-1}, x_n) \ dx_{n-1} $$

$$ + \int s_n (x_{n-1}, x_n) \ p_\theta (x_{n-1} \mid y_{0:n-1}, x_n) \ dx_{n-1}. $$

The integrand in the first equality is $S_n (x_{0:n})$ while the integrand in the first integral of the second equality is $T_{n-1}^\theta (x_{n-1})$. $\blacksquare$

This recursion is not new and is actually a special instance of dynamic programming for Markov processes; see for example \cite{5}. For a fully observed Markov process with transition density $\{ f_\theta (x_k \mid x_{k-1}) \}_{k \geq 1}$, the dynamic programming recursion to compute the expectation of $S_n (x_{0:n})$ with respect to the law of the Markov process is usually implemented using a backward recursion going from time $n$ to time 0. In the partially observed scenario considered here, $\{ X_k \}_{0 \leq k \leq n}$ conditional on $y_{0:n}$ is a “backward” Markov process with non-homogeneous transition densities $\{ p_\theta (x_{k-1} \mid y_{0:k-1}, x_k) \}_{1 \leq k \leq n}$. Thus (2.10) is the corresponding dynamic programming recursion to compute $S_n^\theta$ with respect to $p_\theta (x_{0:n} \mid y_{0:n})$ for this backward Markov chain. This recursion is the foundation of the online EM algorithm and is described at length in \cite{21} (pioneered in \cite{10}) where the density $p_\theta (x_{n-1} \mid y_{0:n-1}, x_n)$ appearing in $T_n^\theta (x_n)$ is usually written as

$$ p_\theta (x_{n-1} \mid y_{0:n-1}, x_n) = \frac{\int f_\theta (x_{n} \mid x_{n-1}) p_\theta (x_{n-1}, y_{0:n-1}) \ dx_{n-1}}{\int f_\theta (x_{n} \mid x_{n-1}) p_\theta (x_{n-1}, y_{0:n-1}) \ dx_{n-1}}$$
or as in Eq. (2.2) in [8], [9], [15]. The forward smoothing recursion has been rediscovered independently several times; see [27], [33] for example.

A simple SMC scheme to approximate $S_n^0$ can be devised by exploiting equations (2.9) and (2.10). This is summarised as Algorithm SMC-FS below.

**Algorithm SMC-FS: Forward-only SMC computation of the FFBS estimate**

- Assume at time $n-1$ that SMC approximations $\left\{ W_{n-1}^{(i)}, X_{n-1}^{(i)} \right\}_{1 \leq i \leq N}$ of $p_0(dx_{n-1} | y_{0:n-1})$
- and $\left\{ \hat{T}_{n-1}^0 \left( X_{n-1}^{(i)} \right) \right\}_{1 \leq i \leq N}$ of $\left\{ T_{n-1}^0 \left( X_{n-1}^{(i)} \right) \right\}_{1 \leq i \leq N}$ are available.
- At time $n$, compute the SMC approximation $\left\{ W_n^{(i)}, X_n^{(i)} \right\}_{1 \leq i \leq N}$ of $p_0(dx_n | y_{0:n})$ and set

\[
\hat{T}_n^0 \left( X_n^{(i)} \right) = \frac{\sum_{j=1}^{N} W_{n-1}^{(j)} f_\theta \left( X_n^{(j)} | X_{n-1}^{(j)} \right) \left[ \hat{T}_{n-1}^0 \left( X_{n-1}^{(j)} \right) + s_n \left( X_{n-1}^{(j)}, X_n^{(i)} \right) \right]}{\sum_{j=1}^{N} W_{n-1}^{(j)} f_\theta \left( X_n^{(j)} | X_{n-1}^{(j)} \right)}
\]

\[
\hat{S}_n^0 = \sum_{i=1}^{N} W_n^{(i)} \hat{T}_n^0 \left( X_n^{(i)} \right).
\]

This algorithm is initialized by setting $\hat{T}_0^0 \left( X_0^{(i)} \right) = 0$ for $1 \leq i \leq N$. It has a computational complexity of $O(N^2)$ which can be reduced by using fast computational methods.

The rationale for this algorithm is as follows. By using $\hat{p}_0(dx_n | y_{0:n-1}, x_n)$ defined in Eq. (2.4) in place of $p_0(dx_n | y_{0:n-1}, x_n)$ in Eq. (2.10), we obtain an approximation $\hat{T}_n^0 (x_n)$ of $T_n^0 (x_n)$ which is computed at the particle locations $\left\{ X_n^{(i)} \right\}_{1 \leq i \leq N}$. The approximation of $S_n^0$ in Eq. (2.12) now follows from Eq. (2.9) by using $\hat{p}_0(dx_n | y_{0:n})$ in place of $p_0(dx_n | y_{0:n})$.

It is valid to use the same notation for the estimates in Eq. (2.7) and in Eq. (2.12) as they are indeed the same. The verification of this assertion may be accomplished by unfolding the recursion in Eq. (2.11).

### 3 Theoretical results

In this section, we present a bound on the non-asymptotic mean square error of the estimate $\hat{S}_n^0$ of $S_n^0$. For sake of simplicity, the result is established for additive functionals of the type

\[
S_n (x_{0:n}) = \sum_{k=0}^{n} s_k (x_k)
\]

where $s_k : \mathcal{X} \to \mathbb{R}$, and when Algorithm SMC-FS is implemented using the bootstrap particle filter; see [7], [19] for a definition of this “vanilla” particle filter. The result can be generalised to accommodate an auxiliary implementation of the particle filter [4], [17], [35]. Likewise, the conclusion is also valid for additive functionals of the type in (1.5); the proof uses similar arguments but is more complicated.
The following regularity condition will be assumed.

(A) There exist constants $0 < \rho, \delta < \infty$ such that for all $x, x' \in \mathcal{X}$, $y \in \mathcal{Y}$ and $\theta \in \Theta$,

$$\rho^{-1} \leq f_\theta(x'|x) \leq \rho, \quad \delta^{-1} \leq g_\theta(y|x) \leq \delta.$$ 

Admittedly, this assumption is restrictive and typically holds when $\mathcal{X}$ and $\mathcal{Y}$ are finite or are compact spaces. In general, quantifying the errors of SMC approximations under weaker assumptions is possible [17]. (More precise but complicated error bounds for the particle estimate of $S_n^\theta$ are also presented in [15] under weaker assumptions.) However, when (A) holds, the bounds can be greatly simplified to the extent that they can usually be expressed as linear or quadratic functions of the time horizon $n$. These simple rates of growth are meaningful as they have also been observed in numerical studies even in scenarios where Assumption A is not satisfied [37].

For a function $s : \mathcal{X} \to \mathbb{R}$, let $\|s\| = \sup_{x \in \mathcal{X}} |s(x)|$. The oscillation of $s$, denoted $\text{osc}(s)$, is defined to be sup $\{|s(x) - s(y)| ; x, y \in \mathcal{X}\}$. The main result in this section is the following non-asymptotic bound for the mean square error of the estimate $\hat{S}_n^\theta$ of $S_n^\theta$ given in Eq. (2.12).

**Theorem 3.1** Assume (A). Consider the additive functional $S_n$ in (3.1) with $\|s_k\| < \infty$ and $\text{osc}(s_k) \leq 1$ for $0 \leq k \leq n$. Then, for any $n \geq 0$ and $\theta \in \Theta$,

$$\begin{align*}
\mathbb{E} \left( \left| \hat{S}_n^\theta - S_n^\theta \right|^2 \right) &\leq a \left( \frac{n+1}{N} \right) \left( 1 + \sqrt{\frac{n+1}{N}} \right)^2 \\
&\text{where } a \text{ is a finite constant that is independent of time } n, \theta \text{ and the particular choice of functions } \{s_k\}_{0 \leq k \leq n}.
\end{align*}$$

The proof is given in the Appendix. It follows that the asymptotic variance of $\sqrt{N}(\hat{S}_n^\theta - S_n^\theta)$, i.e. as the number of particles $N$ goes to infinity, is upper bounded by a quantity proportional to $(n+1)$ as the bias of the estimate is $\mathcal{O}(1/N)$ [15 Corollary 5.3].

Let $\hat{R}_n^\theta$ denote the SMC estimate of $S_n^\theta$ obtained with the standard path space method. This estimate can have a much larger asymptotic variance as is illustrated with the following very simple model. Let $f_\theta(x'|x) = \mu_\theta(x')$, i.e. $\{X_n\}_{n \geq 0}$ is an i.i.d. sequence, and let $y_k = y$ and $s_k = s$ for all $k \geq 1$ where $s$ is some real valued function on $\mathcal{X}$, and $s_0 = 0$. It can be easily established that the formula for the asymptotic variance of $\sqrt{N}(\hat{R}_n^\theta - S_n^\theta)$ given in [11], [14] eqn. (9.13), page 304 | simplifies to

$$n \int \frac{[\pi_\theta(x|y) \tilde{s}_\theta(x)]^2}{\mu_\theta(x)} dx + \frac{n(n-1)}{2} \int \frac{\pi_\theta(x|y)^2}{\mu_\theta(x)} dx \int \tilde{s}_\theta(x)^2 \pi_\theta(x|y) dx$$

where

$$\begin{align*}
\pi_\theta(x|y) &= \frac{\mu_\theta(x) g_\theta(y|x)}{\int \mu_\theta(x') g_\theta(y|x') dx'}, \\
\tilde{s}_\theta(x) &= s(x) - \int s(x) \pi_\theta(x|y) dx.
\end{align*}$$
Thus the asymptotic variance increases quadratically with time \( n \). Note though that the asymptotic variance of \( \sqrt{N} \left( n^{-1} \hat{\mathcal{R}}_n^\theta - n^{-1} \mathcal{S}_n^\theta \right) \) converges as \( n \) tends to infinity to a positive constant. Thus path space method can provide stable estimates of \( \mathbb{E}_\theta [\{ n^{-1} S_n (X_{0:n}) | y_{0:n} \}] \), i.e. when the additive functionals are time-averaged. Let

\[
S_{\gamma,n} (x_{0:n}) = \gamma_n s(x_n) + \sum_{k=1}^{n-1} s(x_k) \gamma_k \prod_{i=k+1}^{n} (1 - \gamma_i)
\]

where \( \{\gamma_n\}_{n \geq 1} \) is a positive non-increasing sequence that satisfies the following constraints: \( \sum_n \gamma_n = \infty \) and \( \sum_n \gamma_n^2 < \infty \). When \( \gamma_n = n^{-1} \) then \( S_{\gamma,n} (x_{0:n}) = n^{-1} S_n (x_{0:n}) \). One important choice for recursive parameter estimation (see Section 4) is

\[
\gamma_n = n^{-\alpha}, \quad 0.5 < \alpha \leq 1. \quad (3.4)
\]

It is also of interest to quantify the stability of the path space method when applied to estimate \( \mathcal{S}_n^\theta = \mathbb{E}_\theta [S_{\gamma,n} (X_{0:n}) | y_{0:n}] \) in this more general time-averaging setting. Once again let \( \hat{\mathcal{R}}_n^\theta \) denote the SMC estimate of \( \mathbb{E}_\theta [S_{\gamma,n} (X_{0:n}) | y_{0:n}] \) obtained with the standard path space method. Using the formula for the asymptotic variance of \( \sqrt{N} (\hat{\mathcal{R}}_n^\theta - \mathcal{S}_n^\theta) \) given in [11], [13] eqn. (9.13), page 304 it can be verified that this asymptotic variance is

\[
\int \frac{[\pi_\theta (x | y) \tilde{s}_\theta (x)]^2}{\mu_\theta (x)} dx \sum_{k=1}^{n} \gamma_k^2 \prod_{i=k+1}^{n} (1 - \gamma_i)^2 + \int \frac{[\pi_\theta (x | y)]^2}{\mu_\theta (x)} dx \int \tilde{s}_\theta (x) dx \sum_{k=2}^{n} \sum_{i=1}^{k-1} \gamma_i^2 (1 - \gamma_{i+1})^2 \cdots (1 - \gamma_n)^2
\]

It follows from Lemma [A.3] in Appendix that any accumulation point of this sequence (in \( n \)) has to be positive. In contrast, the asymptotic variance of \( \sqrt{N} (\hat{\mathcal{S}}_n^\theta - \mathcal{S}_n^\theta) \), i.e. when \( \mathbb{E}_\theta [S_{\gamma,n} (X_{0:n}) | y_{0:n}] \) is computed using Algorithm SMC-FS, will converge to zero as \( n \) tends to infinity.

4 Application to SMC parameter estimation

An important application of the forward smoothing recursion is to parameter estimation for non-linear non-Gaussian SSMs. We will assume that observations are generated from an unknown ‘true’ model with parameter value \( \theta^* \in \Theta \), i.e. \( X_n | (X_{n-1} = x_{n-1}) \sim f_\theta^* (\cdot | x_{n-1}) \) and \( Y_n | (X_n = x_n) \sim g_\theta^* (\cdot | x_n) \). The static parameter estimation problem has generated a lot of interest over the past decade and many SMC techniques have been proposed to solve it; see [28] for a recent review.

4.1 Brief literature review

In a Bayesian approach to the problem, a prior distribution is assigned to \( \theta \) and the sequence of posterior densities \( \{ p (\theta, x_{0:n} | y_{0:n}) \}_{n \geq 0} \) is estimated recursively using SMC algorithms combined with Markov chain Monte Carlo (MCMC) steps [1], [24], [38]. Unfortunately these methods suffer from the particle path degeneracy problem and will result in unreliable
estimates of the model parameters; see [3, 34] for a discussion of this issue. Given a fixed observation record \( y_{0:n} \), an alternative online MCMC approach to estimate \( p(\theta, x_{0:n}|y_{0:n}) \) has been recently proposed which relies on proposals built using the SMC approximation of \( p_\theta(x_{0:n}|y_{0:n}) \) [2].

In a ML approach, the estimate of \( \theta^* \) is the maximising argument of the likelihood of the observed data. The ML estimate can be calculated using a gradient ascent algorithm either offline for a fixed batch of data or online [32], see Section 4.2. Likewise, the EM algorithm can also be implemented offline or online. The online EM algorithm, assuming all calculations can be performed exactly, is presented in [25, 22, 23, 33] and [9]. For a general SSM for which the quantities required by the online EM cannot be calculated exactly, an SMC implementation is possible [3, 28, Section 3.2.]; see Section 4.3.

4.2 Gradient ascent algorithms

To maximise the likelihood \( p_\theta(y_{0:n}) \) w.r.t. \( \theta \), we can use a simple gradient algorithm. Let \( \{\theta_i\}_{i \in \mathbb{N}} \) be the sequence of parameter estimates of the gradient algorithm. We update the parameter at iteration \( i + 1 \) using

\[
\theta_{i+1} = \theta_i + \gamma_{i+1} \nabla \log p_\theta(y_{0:n})|_{\theta=\theta_i}
\]

where \( \nabla \log p_\theta(y_{0:n})|_{\theta=\theta_i} \) is the score vector computed at \( \theta = \theta_i \) and \( \{\gamma_i\}_{i \geq 1} \) is a sequence of positive non-increasing step-sizes defined in (3.4). For a general SSM, we need to approximate \( \nabla \log p_\theta(y_{0:n})|_{\theta=\theta_i} \). As mentioned in the introduction, the score vector admits several smoothed additive functional representations; see Eq. (1.7) and [12]. Using Eq. (1.7), it is possible to approximate the score with Algorithm SMC-FS.

In the online implementation, the parameter estimate at time \( n + 1 \) is updated according to [31, 32]

\[
\theta_{n+1} = \theta_n + \gamma_{n+1} \nabla \log p_{\theta_0,n}(y_n|y_{0:n-1})
\]

Upon receiving \( y_n \), \( \theta_n \) is updated in the direction of ascent of the predictive density of this new observation. A necessary requirement for an online implementation is that the previous values of the model parameter estimates (other than \( \theta_n \)) are also used in the evaluation of \( \nabla \log p_\theta(y_{0:n}|y_{0:n-1}) \) at \( \theta = \theta_n \). This is indicated in the notation \( \nabla \log p_{\theta_0,n}(y_n|y_{0:n-1}) \). (Not doing so would require browsing through the entire history of observations.) This approach was suggested by [32] for the finite state-space case and is named RML. The asymptotic properties of this algorithm (i.e. the behavior of \( \theta_n \) in the limit as \( n \) goes to infinity) have been studied in the case of an i.i.d. hidden process by [39] and for an HMM with a finite state-space by [32]. Under suitable regularity assumptions, convergence to \( \theta^* \) and a central limit theorem for the estimation error has been established.

For a general SSM, we can compute a SMC estimate of \( \nabla \log p_{\theta_0,n}(y_n|y_{0:n-1}) \) using Algorithm SMC-FS upon noting that \( \nabla \log p_{\theta_0,n}(y_n|y_{0:n-1}) \) is equal to

\[
\nabla \log p_{\theta_0,n}(y_{0:n}) - \nabla \log p_{\theta_0,n-1}(y_{0:n-1}).
\]

In particular, at time \( n \), a particle approximation \( \left\{ W_n^{(i)}, X_n^{(i)} \right\}_{1 \leq i \leq N} \) of \( p_{\theta_0,n}(dx_n|y_{0:n}) \) is computed using the particle approximation at time \( n - 1 \) and parameter value \( \theta = \theta_n \). Similarly, the computation of Eq. (2.11) is performed using \( \theta = \theta_n \) and with

\[
s_n(x_{n-1}, x_n) = \nabla \log f_\theta(x_n|x_{n-1})|_{\theta=\theta_n} + \nabla \log g_\theta(y_n|x_n)|_{\theta=\theta_n}.
\]
The estimate of $\nabla \log p_{\theta_0}(y_n|y_{0:n-1})$ is now the difference of the estimate in Eq. (2.12) with the same estimate computed at time $n - 1$.

Under the regularity assumptions given in Section 3, it follows from the results in the Appendix that the asymptotic variance (i.e. as $N \to \infty$) of the SMC estimate of $\nabla \log p_{\theta_0}(y_n|y_{0:n-1})$ computed using Algorithm SMC-FS is uniformly (in time) bounded. On the contrary, the standard path-based SMC estimate of $\nabla \log p_{\theta_0}(y_n|y_{0:n-1})$ has an asymptotic variance that increases linearly with $n$.

4.3 EM algorithms

Gradient ascent algorithms are more generally applicable than the EM algorithm. However, their main drawback in practice is that it is difficult to properly scale the components of the computed gradient vector. For this reason the EM algorithm is usually favoured by practitioners whenever it is applicable.

Let $\{\theta_i\}_{i \in \mathbb{N}}$ be the sequence of parameter estimates of the EM algorithm. In the offline approach, at iteration $i + 1$, the function

$$Q(\theta_i, \theta) = \int \log p_{\theta}(x_{0:n}, y_{0:n}) p_{\theta_i}(x_{0:n}|y_{0:n}) \, dx_{0:n}$$

is computed and then maximized. The maximizing argument is the new estimate $\theta_{i+1}$. If $p_{\theta}(x_{0:n}, y_{0:n})$ belongs to the exponential family, then the maximization step is usually straightforward. We now give an example of this.

Let $s^l : \mathcal{X} \times \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, $l = 1, \ldots, m$, be a collection of functions with corresponding additive functionals

$$S_{l,n}(x_{0:n}, y_{0:n}) = \sum_{k=1}^{n} s^l(x_{k-1}, x_k, y_k), \quad 1 \leq l \leq m,$$

and let

$$S_{l,n}^{\theta} = \int S_{l,n}(x_{0:n}, y_{0:n}) p_{\theta}(x_{0:n}|y_{0:n}) \, dx_{0:n}.$$

The collection $\{S_{l,n}^{\theta}\}_{1 \leq l \leq m}$ is also referred to as the summary statistics in the literature. Typically, the maximising argument of $Q(\theta_i, \theta)$ can be characterised explicitly through a suitable function $\Lambda : \mathbb{R}^m \to \Theta$, i.e.

$$\theta_{i+1} = \Lambda \left( n^{-1} S_{n}^{\theta_i} \right)$$

where $[S_{n}^{\theta}]_l = S_{l,n}^{\theta}$. As an example of this, consider the following stochastic volatility model [35].

Example 4.1 The stochastic volatility model is a SSM defined by the following equations:

$$X_0 \sim \mathcal{N} \left( 0, \frac{\sigma^2}{1 - \phi^2} \right), \quad X_{n+1} = \phi X_n + \sigma V_{n+1},$$

$$Y_n = \beta \exp \left( \frac{X_n}{2} \right) W_n,$$

where $\{V_n\}_{n \in \mathbb{N}}$ and $\{W_n\}_{n \geq 0}$ are independent and identically distributed standard normal noise sequences, which are also independent of each other and of the initial state $X_0$. The
model parameters \( \theta \triangleq (\phi, \sigma^2, \beta^2) \in \mathbb{R} \times (0, \infty) \times (0, \infty) \) are to be estimated. To apply the EM algorithm to this model, let

\[
\begin{align*}
    s^1(x_{n-1}, x_n, y_n) &= x_{n-1}x_n, \\
    s^2(x_{n-1}, x_n, y_n) &= (x_{n-1})^2, \\
    s^3(x_{n-1}, x_n, y_n) &= (x_n)^2, \\
    s^4(x_{n-1}, x_n, y_n) &= y_n^2 \exp(-x_n).
\end{align*}
\]

For large \( n \), we can safely ignore the terms associated to the initial density \( \mu_\theta(x) \) and the solution to the maximisation step is characterised by the function

\[
\Lambda(z_1, z_2, z_3, z_4) = \left( \frac{z_1}{z_2}, z_3 + \left( \frac{z_1}{z_2} \right)^2 z_2 - 2 \left( \frac{z_1}{z_2} \right) z_1, z_4 \right).
\]

The SMC implementation of the forward smoothing recursion has advantages even for the batch EM algorithm. As there is no backward pass, there is no need to store the particle approximations of \( \{p_\theta(dx_k|y_{0:k})\}_{k=0,\ldots,n} \), which can result in a significant memory saving for large data sets.

In the online implementation, running averages of the sufficient statistics are computed instead \[8, 22, 23, 25, 28\] Section 3.2, \[33\]. Let \( \{\theta_k\}_{0 \leq k \leq n} \) be the sequence of parameter estimates of the online EM algorithm computed sequentially based on \( y_{0:n} \). When \( y_{n+1} \) is received, for each \( l = 1, \ldots, m \), compute

\[
S_{l,n+1} = \gamma_{n+1} \int s^l(x_n, x_{n+1}, y_{n+1}) p_{\theta_0,n}(x_n, x_{n+1}|y_{0:n+1}) dx_{n+1} + (1 - \gamma_{n+1}) \sum_{k=1}^n \prod_{i=k+1}^n (1 - \gamma_i) \gamma_k s^l(x_{k-1}, x_k, y_k) p_{\theta_0,n}(x_n, y_{0:n+1}) dx_{0:n},
\]

and then set

\[
\theta_{n+1} = \Lambda(S_{n+1})
\]

where \([S_{n+1}] = S_{l,n+1}\). Here \( \{\gamma_n\}_{n \geq 1} \) is a step-size sequence satisfying the same conditions stipulated for the RML in Section 12.2. The recursive implementation of \( S_{l,n+1} \) is standard \[4\]. The subscript \( \theta_{0:n} \) on \( p_{\theta_0,n}(x_n|y_{0:n}) \) indicates that the posterior density is being computed sequentially using the parameter \( \theta_{k-1} \) at time \( k \) (and \( \theta_0 \) at time 0). References \[8, 22, 23\] chapter 4 and \[33\] have proposed an online EM algorithm, implemented as above, for finite state HMMs. In the finite state setting all computations involved can be done exactly in contrast to general SSMs where numerical procedures are called for. It is also possible to do all the calculations exactly for linear Gaussian models \[23\].

Define the vector valued function \( s : \mathcal{X} \times \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^m \) as follows: \( s = [s^1, \ldots, s^m]^T \).

Computing \( S_n \) sequentially using SMC-FS is straightforward and detailed in the following algorithm.

SMC-FS implementation of online EM

At time \( n = 0 \)

- Choose \( \theta_0 \).
- Set \( T_0^{(i)} = 0 \in \mathbb{R}^m, i = 1, \ldots, N \).
- Construct the SMC approximation \( \{X_0^{(i)}, W_0^{(i)}\}_{1 \leq i \leq N} \) of \( p_{\theta_0}(dx_0|y_0) \).
At times \( n \geq 1 \)
- Construct the SMC approximation \( \{X_n^{(i)}, W_n^{(i)}\}_{1 \leq i \leq N} \) of \( p_{\theta_n} (dx_n | y_{0:n}) \).
- For each \( i = 1, \ldots, N \), compute
  \[
  T_n^{(i)} = \frac{\sum_{j=1}^{N} W_{n-1}^{(j)} f_{\theta_{n-1}} (X_n^{(i)} | X_{n-1}^{(j)}) \left[ (1 - \gamma_n) T_{n-1}^{(j)} + \gamma_n \Phi (X_n^{(j)}, X_n^{(i)}, y_n) \right]}{\sum_{j=1}^{N} W_{n-1}^{(j)} f_{\theta_{n-1}} (X_n^{(i)} | X_{n-1}^{(j)})}.
  \]
- Compute \( \hat{S}_n = \sum_{i=1}^{N} W_n^{(i)} T_n^{(i)} \) and update the parameter, \( \theta_n = \Lambda (\hat{S}_n) \).

It was suggested in [28 Section 3.2.] that the two other SMC methods discussed in Section 1.2 could be used to approximate \( S_n \); the path space approach to implement the online EM was also independently proposed in [8]. Doing so would yield a cheaper alternative to Algorithm SMC-EM above with computational cost \( O (N) \), but not without its drawbacks. The fixed-lag approximation of [29] would introduce a bias which might be difficult to control and the path space approach suffers from the usual particle path degeneracy problem. Consider the step-size sequence in (3.4). If the path space method is used to estimate \( S_n \) then the theory in Section 3 tells us that, even under strong mixing assumptions, the asymptotic variance of the estimate of \( S_n \) will not converge to zero for \( 0.5 < \alpha \leq 1 \). Thus it will not yield a theoretically convergent algorithm. Numerical experiments in [8] appear to provide stable results which we attribute to the fact that this variance might be very small in the scenarios considered. In contrast, the asymptotic variance of the \( O \left( N^2 \right) \) estimate converges to zero in time \( n \) for the entire range \( 0.5 < \alpha \leq 1 \) under the same mixing conditions. The original \( O \left( N^2 \right) \) implementation proposed here has been recently successfully adopted in [31] to solve a complex parameter estimation problem arising in robotics.

5 Simulations

5.1 Comparing SMC-FS with the path space method

We commence with a study of a scalar linear Gaussian SSM for which we may calculate smoothed functionals analytically. We use these exact values as benchmarks for the SMC approximations. The model is
\[
X_0 \sim \mathcal{N} (0, \sigma_0^2), \quad X_{n+1} = \phi X_n + \sigma V W_{n+1}, \quad \sigma V = c X_n + \sigma W W_n, \tag{5.1}
\]
where \( V_n \overset{i.i.d.}{\sim} \mathcal{N} (0, 1) \) and \( W_n \overset{i.i.d.}{\sim} \mathcal{N} (0, 1) \). We compared the exact values of the following smoothed functionals
\[
S_{1,n}^\theta = \mathbb{E}_{\theta} \left[ \sum_{k=1}^{n} X_k^2 | y_{0:n} \right], \quad S_{2,n}^\theta = \mathbb{E}_{\theta} \left[ \sum_{k=1}^{n} X_k | y_{0:n} \right], \quad S_{3,n}^\theta = \mathbb{E}_{\theta} \left[ \sum_{k=1}^{n} X_{k-1} X_k | y_{0:n} \right],
\]
\[
\text{In a Bayesian framework where } \theta \text{ is assigned a prior distribution and we estimate } \{p (x_n, \theta | y_{0:n})\}_{n \geq 0} \text{ [1, 24, 28], the path degeneracy problem has much more severe consequences than in the ML framework considered here as illustrated in [3]. Indeed in the ML framework, the filter } \{p (x_n | y_{0:n})\}_{n \geq 0} \text{ will have, under regularity assumptions, exponential forgetting properties for any } \theta \in \Theta \text{ whereas this will never be the case for } p (x_n, \theta | y_{0:n}).
\]
computed at \( \theta^* = (\phi^*, \sigma^*_Y, c^*, \sigma^*_W) = (0.8, 0.1, 1.0, 1.0) \) with the bootstrap filter implementation of Algorithm SMC-FS and the path space method. Comparisons were made after 2500, 5000, 7500 and 10,000 observations to monitor the increase in variance and the experiment was replicated 50 times to generate the box-plots in Figure 1. (All replications used the same data record.) Both estimates were computed using \( N = 500 \) particles.

From Figure 1 it is evident that the SMC estimates of Algorithm SMC-FS significantly outperforms the corresponding SMC estimates of the path space method. However one should bear in mind that the former algorithm has \( O(N^2) \) computational complexity while the latter is \( O(N) \). Thus a comparison that takes this difference into consideration is important. From Theorem 3.1 and the discussion after it, we expect the variance of Algorithm SMC-FS’s estimate to grow only linearly with the time index compared to a quadratic in time growth of variance for the path space method. Hence, for the same computational effort we argue that, for large observation records, the estimate of Algorithm SMC-FS is always going to outperform the path space estimates. Specifically, for a large enough \( n \), the variance of Algorithm SMC-FS’s estimate with \( N \) particles will be significantly less than
the variance of the path space estimate with \( N^2 \) particles. If the number of observations is small then, taking into account the computational complexity, it might be better to use the path space estimate as the variance benefit of using Algorithm SMC-FS may not be appreciable to justify the increased computational load.

5.2 Online EM

Figure 2 shows the parameter estimates obtained using the SMC implementation of online EM for the stochastic volatility model discussed in Example 4.1. The true value of the parameters were \( \theta^* = (\phi, \sigma^2, \beta^2) = (0.8, 0.1, 1) \) and 500 particles were used. SMC-EM was started at the initial guess \( \theta_0 = (0.1, 1, 2) \). For the first 100 observations, only the E-step was executed. That is the step \( \theta_n = \Lambda(\hat{S}_n) \), which is the M-step was skipped. SMC-EM was run in its entirety for observations 101 and onwards. The step size used was \( \gamma_n = 0.01 \) for \( n \leq 10^5 \) and \( 1/((n - 5 \times 10^4)^{0.6}) \) for \( n > 10^5 \). Figure 2 shows the sequence of parameter estimates computed with a very long observation sequence.

![Figure 2: Estimating the parameters of the stochastic volatility model with the SMC version of online EM, Algorithm SMC-EM. Initial parameter guess \( \theta_0 = (0.1, 1, 2) \). True and converged values (average of the last 1000 iterations) are indicated on the left and the right of the plot respectively.](image)

6 Discussion

We proposed a new SMC algorithm to compute the expectation of additive functionals recursively in time. Essentially, it is an online implementation of the FFBS SMC algorithm proposed in [18]. This algorithm has an \( \mathcal{O}(N^2) \) computational complexity where \( N \) is the number of particles. It was mentioned how a standard path space SMC estimator to compute the same expectations recursively in time could be developed. This would have an \( \mathcal{O}(N) \) computational complexity. However, as conjectured in [37], it was shown here that the asymptotic variance of the SMC-FFBS estimator increased linearly with time whereas that
of the $O(N)$ method increased quadratically. The online SMC-FFBS estimator was then used to perform recursive parameter estimation. While the convergence of RML and online EM have been established when they can be implemented exactly, the convergence of the SMC implementation of these algorithms have yet to be established and is currently under investigation.

7 Acknowledgments

The authors would like to thank Olivier Cappé, Thomas Flury, Sinan Yildirim and Éric Moulines for comments and references that helped improve the first version of this paper. The authors are also grateful to Rong Chen for pointing out the link between the forward smoothing recursion and dynamic programming. Finally, we are thankful to Robert Elliott to have pointed out to us references [22], [23] and [25].

A Appendix

The proofs in this section hold for any fixed $\theta$ and therefore $\theta$ is omitted from the notation. This section commences with some essential definitions.

Consider the measurable space $(E, \mathcal{E})$. Let $\mathcal{M}(E)$ denote the set of all finite signed measures and $\mathcal{P}(E)$ the set of all probability measures on $E$. Let $\mathcal{B}(E)$ denote the Banach space of all bounded and measurable functions $f$ equipped with the uniform norm $\|f\|$. Let $\nu(f) = \int \nu(dx) f(x)$, i.e. $\nu(f)$ is the Lebesgue integral of the function $f \in \mathcal{B}(E)$ w.r.t. the measure $\nu \in \mathcal{M}(E)$. If $\nu$ is a density w.r.t. some dominating measure $dx$ on $E$ then, $\nu(f) = \int \nu(x) f(x)dx$. We recall that a bounded integral kernel $M(x, dx')$ from a measurable space $(E, \mathcal{E})$ into an auxiliary measurable space $(E', \mathcal{E}')$ is an operator $f \mapsto M(f)$ from $\mathcal{B}(E')$ into $\mathcal{B}(E)$ such that the functions

$$x \mapsto M(f)(x) := \int_{E'} M(x, dx') f(x')$$

are $\mathcal{E}$-measurable and bounded, for any $f \in \mathcal{B}(E')$. In the above displayed formulae, $dx'$ stands for an infinitesimal neighborhood of a point $x'$ in $E'$. Let $\beta(M)$ denote the Dobrushin coefficient of $M$ which defined by the following formula

$$\beta(M) := \sup \{\text{osc}(M(f)) ; \ f \in \text{Osc}_1(E')\}$$

where $\text{Osc}_1(E')$ stands the set of $\mathcal{E}'$-measurable functions $f$ with oscillation less than or equal to 1. The kernel $M$ also generates a dual operator $\nu \mapsto \nu M$ from $\mathcal{M}(E)$ into $\mathcal{M}(E')$ defined by $(\nu M)(f) := \nu(M(f))$. A Markov kernel is a positive and bounded integral operator $M$ with $M(1) = 1$. Given a pair of bounded integral operators $(M_1, M_2)$, we let $(M_1 M_2)$ the composition operator defined by $(M_1 M_2)(f) = M_1(M_2(f))$. For time homogenous state spaces, we denote by $M^m (= M^{m-1} M = MM^{m-1})$ the $m$-th composition of a given bounded integral operator $M$, with $m \geq 1$.

Given a positive function $G$ on $E$, let $\Psi_G : \nu \in \mathcal{P}(E) \mapsto \Psi_G(\nu) \in \mathcal{P}(E)$ be the Bayes transformation defined by

$$\Psi_G(\nu)(dx) := \frac{1}{\nu(G)} G(x) \nu(dx)$$

17
The definitions above also apply if \( \nu \) is a density and \( M \) is a transition density. In this case all instances of \( \nu(dx) \) should be replaced with \( \nu(x)dx \) and \( M(x, dx') \) by \( M(x, x')dx' \) where \( dx \) and \( dx' \) are the dominating measures.

The proofs below will apply to any fixed sequence of observation \( \{y_n\}_{n \geq 0} \) and it is convenient to introduce the following transition kernels,

\[
Q_n(x_{n-1}, dx_n) = g(y_{n-1}|x_{n-1})f(x_n|x_{n-1})dx_n, \quad n \geq 1, \\
\text{and} \quad Q_{k,n} = Q_{k+1}Q_{k+2} \cdots Q_n, \quad 0 \leq k \leq n,
\]

with the convention that \( Q_{n,n} = Id \), the identity operator. Note that \( Q_{k,n}(1) = p(y_{k-1}|x_k) \).

Let the mapping \( \Phi_k : \mathcal{P}(\mathcal{X}) \rightarrow \mathcal{P}(\mathcal{X}) \), \( k \geq 1 \), be defined as follows

\[
\Phi_k(\nu)(dx_k) = \frac{\nu Q_k(dx_k)}{\nu Q_k(1)}.
\]

Several probability densities and their SMC approximations are introduced to simplify the exposition. The predicted filter is denoted by

\[
\eta_n(dx_n) = p(dx_n | y_{0:n-1})
\]

with the understanding that \( \eta_0(dx_0) \) is the initial distribution of \( X_0 \). Let \( \eta_n^N \) denote its SMC approximation with \( N \) particles. (This notation for the SMC approximation is opted for, instead of the usual \( \hat{\eta}_n \), to make the number of particles explicit.) The bounded integral operator \( D_{k,n} \) from \( \mathcal{X} \) into \( \mathcal{X}^{n+1} \) is defined as

\[
D_{k,n}(S_n)(x_k) := \int p(dx_{0:k-1} | x_k, y_{0:k-1}) \left( \prod_{q=k}^{n-1} g(y_q|x_q)f(x_{q+1}|x_q) \right) S_n(x_{0:n}) dx_{k+1:n} \tag{A.1}
\]

\( D_{k,n} \) is defined for any pair of time indices \( k, n \) satisfying \( 0 \leq k \leq n \) with the convention that \( p(x_{0:k-1} | x_k, y_{0:k-1}) = 1 \) for \( k = 0 \) and \( \prod \emptyset = 1 \). The SMC approximation, \( D_{k,n}^N \), is

\[
D_{k,n}^N(S_n)(x_k) := \int p^N(dx_{0:k-1} | x_k, y_{0:k-1}) \left( \prod_{q=k}^{n-1} g(y_q|x_q)f(x_{q+1}|x_q) \right) S_n(x_{0:n}) dx_{k+1:n} \tag{A.2}
\]

where \( p^N(dx_{0:k-1} | x_k, y_{0:k-1}) \) is the SMC approximation of \( p(dx_{0:k-1} | x_k, y_{0:k-1}) \) obtained from the SMC-FFBS approximation of Section 2.1, i.e.

\[
p^N(dx_{0:k-1} | x_k, y_{0:k-1}) = \prod_{q=1}^{k} M_{q, \eta_{q-1}^N}(x_q, dx_{q-1}) \tag{A.3}
\]

where the backward Markov transition kernels \( M_{q, \eta_{q-1}^N} \) are defined through

\[
M_{q, \eta_{q-1}^N}(x_q, dx_{q-1}) = \frac{\eta_{q-1}^N(dx_{q-1})g(y_{q-1}|x_{q-1})f(x_q|x_{q-1})}{\eta_{q-1}^N(g(y_{q-1} | \cdot)f(x_q | \cdot))}. \tag{A.4}
\]

It is easily established that the SMC-FFBS approximation of \( p(dx_k | y_{0:n}) \), \( k \leq n \), is precisely the marginal of

\[
p^N(dx_{0:n-1} | x_n, y_{0:n-1}) \hat{p}(dx_n | y_{0:n})
\]
where \( \tilde{p}(dx_n|y_{0:n}) \) was defined in (1.9). Finally, we define

\[
P_{k,n} = \frac{D_{k,n}}{D_{k,n}(1)} \quad \text{and} \quad P_{k,n}^N = \frac{D_{k,n}^N}{D_{k,n}^N(1)}
\]

The following estimates are a straightforward consequence of Assumption (A). For time indices \( 0 \leq k \leq q \leq n \),

\[
b_{k,n} = \sup_{x_k, x'_k} \frac{Q_{k,n}(1)(x_k)}{Q_{k,n}(1)(x'_k)} \leq \rho^2 \delta^2, \quad \beta \left( \frac{Q_{k,q}(x_k, dx_q)Q_{q,n}(1)(x_q)}{Q_{k,q}(Q_{q,n}(1))} \right) \leq (1 - \rho^{-4})^{(q-k)} \quad (A.5)
\]

and for \( 0 < k \leq q \),

\[
M_{k,\eta}(x, dz) \leq \rho^4 M_{k,\eta}(x', dz) \implies \beta \left( M_{q,\eta_{k-1}^N} \ldots M_{k,\eta_{k-1}^N} \right) \leq (1 - \rho^{-4})^{q-k+1} \quad (A.6)
\]

Several auxiliary results are now presented. For any \( \varphi \in B(\mathcal{X}) \), let

\[
V_k^N(\varphi) = \eta_k^N(\varphi) - \Phi_k(\eta_{k-1}^N)(\varphi) \quad (A.7)
\]

The following is an almost sure Kintchine type inequality [14, Lemma 7.3.3].

**Lemma A.1** Let \( \mathcal{F}^N_n := \sigma \left( \left\{ X_k^{(i)}; 0 \leq k \leq n, 1 \leq i \leq N_n \right\} \right), n \geq 0 \), be the natural filtration associated with the \( N \)-particle approximation model and \( \mathcal{F}^N_{k-1} \) be the trivial sigma field. For any \( r \geq 1 \), there exist a finite (non random) constant \( a_r \) such that the following inequality holds for all \( k \geq 0 \) and \( \mathcal{F}^N_{k-1} \) measurable functions \( \varphi_k^N \in B(\mathcal{X}) \) s.t. osc(\( \varphi_k^N \)) \leq 1,

\[
E \left( \left| \sqrt{N}V_k^N(\varphi_k^N) \right|^r \bigg| \mathcal{F}^N_{k-1} \right) \leq a_r.
\]

This inequality may be used to derive the following \( L_r \) error estimate [14, Theorem 7.4.4].

**Lemma A.2** For any \( r \geq 1 \), there exists a constant \( a_r \) such that the following inequality holds for all \( k \geq 0 \) and \( \varphi \in B(\mathcal{X}) \) s.t. osc(\( \varphi \)) \leq 1,

\[
\sqrt{N}E \left( \left| \eta_k^N - \eta_n \right|(\varphi) \right)^{\frac{r}{2}} \leq a_r \sum_{k=0}^n b_{k,n} \beta \left( \frac{Q_{k,n}}{Q_{k,n}(1)} \right). \quad (A.8)
\]

A time-uniform bound for (A.8) may be obtained by using the estimates in (A.5)- (A.6). The final auxiliary result is the following.

**Lemma A.3** For time indices \( 1 \leq k \leq n \),

\[
\eta_{k-1} D_{k-1,n}^N(S_n) = (\eta_{k-1}^N Q_k)(D_{k,n}^N(S_n))
\]

19
Proof:

\[
\eta^N_{k-1}D^N_{k-1,n}(S_n) = \int \eta^N_{k-1}(dx_{k-1})p^N(dx_{0:k-2}|x_{k-1},y_{0:k-2}) \left( \prod_{q=k}^n Q_q(x_{q-1},dx_q) \right) S_n(x_{0:n})
\]

\[
= \int \eta^N_{k-1}(dx_{k-1})Q_k(x_{k-1},dx_k)
\times p^N(dx_{0:k-2}|x_{k-1},y_{0:k-2}) \left( \prod_{q=k+1}^n Q_q(x_{q-1},dx_q) \right) S_n(x_{0:n})
\]

The result follows upon noting that

\[
\eta^N_{k-1}(dx_{k-1})Q_k(x_{k-1},dx_k) = \eta^N_{k-1}Q_k(dx_k) M_{k,\eta^N_{k-1}}(x_k, dx_{k-1}).
\]

To prove Theorem 3.1 the same semigroup techniques of [13, Section 7.4.3] are employed.

Proof:

The following decomposition is central

\[
\tilde{S}_n - S_n = \sum_{0 \leq k \leq n} \left( \frac{\eta^N_k D^N_{k,n}(S_n)}{\eta^N_k D^N_{k,n}(1)} - \frac{\eta^N_{k-1} D^N_{k-1,n}(S_n)}{\eta^N_{k-1} D^N_{k-1,n}(1)} \right)
\]

with the convention that \(\eta^N_{-1} D^N_{-1,n} = \eta_0(dx_0) \prod_{q=1}^n Q_q(x_{q-1},dx_q)\), for \(k = 0\). Lemma A.3 states that

\[
\eta^N_{k-1}D^N_{k-1,n}(S_n) = (\eta^N_{k-1}Q_k)(D^N_{k,n}(S_n))
\]

and therefore the decomposition can be also written as

\[
\tilde{S}_n - S_n = \sum_{0 \leq k \leq n} \left( \frac{\eta^N_k D^N_{k,n}(S_n)}{\eta^N_k D^N_{k,n}(1)} - \frac{\Phi_k(\eta^N_{k-1})(D^N_{k,n}(S_n))}{\Phi_k(\eta^N_{k-1})(D^N_{k,n}(1))} \right) \tag{A.9}
\]

with the convention \(\Phi_0(\eta^N_{-1}) = \eta_0\), for \(k = 0\). Let

\[
\tilde{S}^N_{k,n} = S_n - \frac{\Phi_k(\eta^N_{k-1})(D^N_{k,n}(S_n))}{\Phi_k(\eta^N_{k-1})(D^N_{k,n}(1))}.
\]

Then every term in the r.h.s. of \(A.9\) takes the following form

\[
\frac{\eta^N_k D^N_{k,n}(\tilde{S}^N_{k,n})}{\eta^N_k D^N_{k,n}(1)} = \frac{\eta^N_k Q_{k,n}(1)}{\eta^N_k Q_{k,n}(1)} \times V_k^N \left( \bar{D}^N_{k,n}(\tilde{S}^N_{k,n}) \right) \tag{A.10}
\]

where the integral operators \(\bar{D}^N_{k,n}\) are defined as follows,

\[
\bar{D}^N_{k,n}(S_n) = \frac{D^N_{k,n}(S_n)}{\eta^N_k Q_{k,n}(1)}.
\]

Finally, using \(A.9\) and \(A.10\), \(\tilde{S}_n - S_n\) is expressed as

\[
\sqrt{N} \left( \tilde{S}_n - S_n \right) = I^N_n(S_n) + R^N_n(S_n)
\]

20
where the first order term is

$$I_n^N(S_n) := \sum_{0 \leq k \leq n} \sqrt{N} V_k^N \left( \overline{D}_{k,n}^N(\tilde{S}_{k,n}) \right)$$

and the second order remainder term is

$$R_n^N(S_n) := \sum_{0 \leq k \leq n} \frac{1}{\eta_k^N \overline{D}_{k,n}^N(1)} \sqrt{N} \left( \eta_k - \eta_k^N \right) \overline{D}_{k,n}^N(1) \times V_k^N \left( \overline{D}_{k,n}^N(\tilde{S}_{k,n}) \right).$$

The non-asymptotic variance bound is based on the triangle inequality

$$\mathbb{E} \left\{ N \left( \tilde{S}_n - S_n \right)^2 \right\} \leq \left( \mathbb{E} \left\{ I_n^N(S_n)^2 \right\} \right)^{\frac{1}{2}} + \left( \mathbb{E} \left\{ R_n^N(S_n)^2 \right\} \right)^{\frac{1}{2}}, \quad (A.11)$$

and bounds are derived below for the individual expressions on the right-hand side of this equation.

Using the fact that $$\left\{ V_k^N \left( \overline{D}_{k,n}^N(\tilde{S}_{k,n}) \right) \right\}_{0 \leq k \leq n}$$ is zero mean and uncorrelated,

$$\mathbb{E} \left( I_n^N(S_n)^2 \right) = \sum_{0 \leq k \leq n} N \mathbb{E} \left\{ V_k^N \left( \overline{D}_{k,n}^N(\tilde{S}_{k,n}) \right)^2 \right\}. \quad (A.12)$$

The following results are needed to bound the right-hand side of (A.12). First, observe that $$D_{k,n}^N(1) = Q_{k,n}(1),$$ and $$\overline{D}_{k,n}^N(1) = \overline{D}_{k,n}(1).$$ Now using the decomposition,

$$\overline{D}_{k,n}^N(\tilde{S}_{k,n})(x_k) = \overline{D}_{k,n}(1)(x_k) \times \int \left[ P_{k,n}^N(S_n)(x_k) - P_{k,n}^N(S_n)(x'_k) \right] \Psi_{k,n}(1) \Phi_k(\eta_k)(dx'_k),$$

it follows that

$$\left\| \overline{D}_{k,n}^N(\tilde{S}_{k,n}) \right\| \leq b_{k,n} \text{osc}(P_{k,n}^N(S_n)) \quad (A.13)$$

For linear functionals of the form (3.1), it is easily checked that

$$D_{k,n}^N(S_n) = Q_{k,n}(1) \sum_{0 \leq q \leq k} \left[ M_{k,\eta_{q-1}^N} \cdots M_{q+1,\eta_q^N} \right] (s_q) + \sum_{k < q \leq n} Q_{k,q}(s_q) Q_{q,n}(1)$$

with the convention $$M_{k,\eta_{q-1}^N} \cdots M_{k+1,\eta_q^N} = Id, the identity operator, for q = k. Recalling that D_{k,n}^N(1) = Q_{k,n}(1), we conclude that

$$P_{k,n}^N(S_n) = s_k + \sum_{0 \leq q < k} \left[ M_{k,\eta_{q-1}^N} \cdots M_{q+1,\eta_q^N} \right] (s_q) + \sum_{k < q \leq n} Q_{k,q}(Q_{q,n}(1) \frac{s_q}{Q_{k,q}(Q_{q,n}(1))})$$

and therefore

$$P_{k,n}^N(S_n) = \sum_{0 \leq q < k} \left[ M_{k,\eta_{q-1}^N} \cdots M_{q+1,\eta_q^N} \right] (s_q) + \sum_{k \leq q \leq n} \frac{Q_{k,q}(Q_{q,n}(1) \frac{s_q}{Q_{k,q}(Q_{q,n}(1))})}{Q_{k,q}(Q_{q,n}(1))}$$
Thus,
\[
\text{osc}(P_{k,n}^N(S_n)) \\
\leq \sum_{0 \leq q < k} \beta \left( M_{k,q} \eta_{q-1}^N \cdots M_{q+1} \eta_N^N \right) \text{osc}(s_q) + \sum_{k \leq q \leq n} \beta \left( Q_{k,q}(x_k, dx_q)Q_{q,n}(1)(x_q) \right) \text{osc}(s_q) \\
(\text{A.14})
\]

Using the estimates in (A.5) and (A.6) for the contraction coefficients, and the estimate in (A.5) for \( b_{k,n} \), it follows that there exists some finite (non random) constant \( c \) such that the bound
\[
\|D_{k,n}^N(\tilde{S}_{k,n}^N)\| \leq c \\
(\text{A.15})
\]
holds for any pair of time indexes \( k, n \) satisfying \( 0 \leq k \leq n \), particle number \( N \) and choice of functions \( \{s_k\}_{0 \leq k \leq n} \). The desired bound for (A.14) is now obtained by combining this result with Lemma A.1:

\[
\begin{align*}
E\left(I_n^N(S_n)^2\right) &= \sum_{0 \leq k \leq n} N^2 \mathbb{E}\left(V_{k,n}^N(D_{k,n}^N(\tilde{S}_{k,n}^N))^2\right) \\
&\leq d(n + 1) \\
(\text{A.16})
\end{align*}
\]

where \( d \) is a constant whose value does not depend on \( (n, N, S_n) \). Concerning the term \( \mathbb{E}\{R_{k,n}^N(S_n)^2\} \) in (A.11),

\[
\begin{align*}
\mathbb{E}\{R_{n}^N(S_n)^2\}^{\frac{1}{2}} &\leq \sum_{0 \leq k \leq n} \frac{1}{\sqrt{N}} \mathbb{E}\left\{ \left[ \frac{1}{\eta_k} D_{k,n}(1) \sqrt{N} (\eta_k - \eta_k^N) D_{k,n}(1) \times \sqrt{N} V_{k,n}^N \left( D_{k,n}^N(\tilde{S}_{k,n}^N) \right) \right]^2 \right\}^{\frac{1}{2}} \\
&\leq \sum_{0 \leq k \leq n} \frac{1}{\sqrt{N}} b_{k,n} \mathbb{E}\left[ \left[ \sqrt{N} (\eta_k - \eta_k^N) D_{k,n}(1) \times \sqrt{N} V_{k,n}^N \left( D_{k,n}^N(\tilde{S}_{k,n}^N) \right) \right]^2 \right]^{\frac{1}{2}} \\
&\leq \sum_{0 \leq k \leq n} \frac{1}{\sqrt{N}} b_{k,n} \mathbb{E}\left[ \left[ \sqrt{N} V_{k,n}^N \left( D_{k,n}^N(\tilde{S}_{k,n}^N) \right) \right]^4 \right]^{\frac{1}{4}} \\
&\times \mathbb{E}\left[ \left[ \sqrt{N} V_{k,n}^N \left( D_{k,n}^N(\tilde{S}_{k,n}^N) \right) \right]^4 \right]^{\frac{1}{4}} \\
&\leq \frac{1}{\sqrt{N}} e(n + 1) \\
(\text{A.17})
\end{align*}
\]

where \( e \) is a constant whose value does not depend on \( (n, N, S_n) \). The second line follows from (A.5) and the third by the Cauchy-Schwartz inequality. The final line was arrived at by the same reasoning used to derive bound (A.16) and Lemma A.2. The assertion of the theorem may be verified by substituting bounds (A.16) and (A.17) into (A.11).

It is possible to write
\[
\sum_{k=1}^{n} \gamma_k^2 \prod_{i=k+1}^{n} (1 - \gamma_i)^2 + \sum_{k=2}^{n} \sum_{i=1}^{k-1} \gamma_i^2 (1 - \gamma_{i+1})^2 \cdots (1 - \gamma_n)^2
\]
as the sum in (A.18) below.
Lemma A.4 Let $\alpha \in (0.5, 1]$ and $\gamma_n = n^{-\alpha}$ for $n > 0$. Then

$$\lim \inf_{n \to \infty} \gamma_n^2 + \sum_{i=1}^{n-1} (n + 1 - i)\gamma_i^2(1 - \gamma_{i+1})^2 \cdots (1 - \gamma_n)^2 > 0.$$  \hfill (A.18)

Proof:
Let $\lfloor a \rfloor$ denote the largest integer less than or equal to $a$. Since the result is obvious for $\alpha = 1$, let $\alpha \in (0.5, 1)$.

$$\gamma_n^2 + \sum_{i=\lfloor n/2 \rfloor}^{n-1} (n + 1 - i)\gamma_i^2(1 - \gamma_{i+1})^2 \cdots (1 - \gamma_n)^2$$

$$\geq \gamma_n^2 + \gamma_n^2 \sum_{i=\lfloor n/2 \rfloor}^{n-1} (n + 1 - i)(1 - \gamma_{\lfloor n/2 \rfloor})^{2(n-i)}$$

$$= \gamma_n^2 \left( \sum_{j=1}^{n+1-\lfloor n/2 \rfloor} j\lambda_n^{j-1} - \frac{1}{(1 - \lambda_n)^2} \right) + \frac{\gamma_n^2}{(1 - \lambda_n)^2}$$

where $\lambda_n = (1 - \gamma_{\lfloor n/2 \rfloor})^2$ and

$$\sum_{j>0} j\lambda_n^{j-1} = \frac{1}{(1 - \lambda_n)^2}.$$

It may be verified that

$$\lim_{n \to \infty} \frac{\gamma_n^2}{(1 - \lambda_n)^2} = 2^{-2\alpha - 2}$$

and

$$\lim_{n \to \infty} \gamma_n^2 \sum_{j>n+1-\lfloor n/2 \rfloor} j\lambda_n^{j-1} = 0.$$

Hence the result follows.

References

[1] Andrieu, C., De Freitas, J.F.G. and Doucet, A. (1999). Sequential MCMC for Bayesian model selection. Proc. IEEE Workshop on Higher Order Statistics, 130–134.

[2] Andrieu, C., Doucet, A. and Holenstein, R. (2010). Particle Markov chain Monte Carlo (with discussion). J. Royal Statist. Soc. B, 72, 269-342.

[3] Andrieu, C., Doucet, A. and Tadić, V. B. (2005). On-line parameter estimation in general state-space models. Proc. 44th IEEE Conf. on Decision and Control, 332–337.

[4] Benveniste, A., Métivier, M. and Priouret, P. (1990). Adaptive Algorithms and Stochastic Approximation. New York: Springer-Verlag.

[5] Bertsekas, D. and Shreve, S.E. (1978) Stochastic Optimal Control: The Discrete-Time Case, Academic Press.
[6] Carpenter, J., Clifford, P. and Fearnhead, P. (1999). An improved particle filter for non-linear problems. *IEE proceedings - Radar, Sonar and Navigation*, **146**, 2-7.

[7] Cappé, O., Moulines, É. and Rydén, T. (2005) *Inference in Hidden Markov Models*. New York: Springer-Verlag.

[8] Cappé, O. (2009). Online sequential Monte Carlo EM algorithm. *Proc. IEEE Workshop on Statistical Signal Processing*, Cardiff, UK.

[9] Cappé, O. (2009). Online EM algorithm for hidden Markov models. Available at [http://arxiv.org/abs/0908.2359](http://arxiv.org/abs/0908.2359)

[10] Cérou, F., Del Moral, P. and Guyader, A. (2008). A non asymptotic variance theorem for unnormalized Feynman-Kac particle models, Technical report INRIA-00337392. Available at [http://hal.inria.fr/inria-00337392-v1/](http://hal.inria.fr/inria-00337392-v1/)

[11] Chopin, N. (2004). Central limit theorem for sequential Monte Carlo and its application to Bayesian inference. *Ann. Statist.*, **32**, 2385-2411.

[12] Coquelin, P.A., Deguest, R. and Munos, R. (2009). Sensitivity analysis in HMMs with application to likelihood maximization. *Proc. Conf. NIPS*, Vancouver, Canada.

[13] Del Moral, P. and Doucet, A. (2003). On a class of genealogical and interacting Metropolis models. *Lecture Notes in Mathematics* **1832**, Berlin: Springer-Verlag, 415-46.

[14] Del Moral, P. (2004). *Feynman-Kac Formulae: Genealogical and Interacting Particle Systems with Applications*. New York: Springer-Verlag.

[15] Del Moral, P., Doucet, A. and Singh. S.S. (2010). A backward particle interpretation of Feynman-Kac formulae. *ESAIM: Math. Model. Num. Analy.*, **44**, 947-975.

[16] Del Moral, P. and Guionnet, A. (2001). On the stability of interacting processes with applications to filtering and genetic algorithms. *Ann. Inst. H. Poincaré Probab. Statist.*, **37**, 155–194.

[17] Douc, R., Garivier, A., Moulines, E. and Olsson, J. (2011). Sequential Monte Carlo smoothing for general state space hidden Markov models. *Ann. Applied Proba.*, to appear.

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

[19] Doucet, A., De Freitas, J.F.G. and Gordon N.J. (eds.) (2001). *Sequential Monte Carlo Methods in Practice*. New York: Springer-Verlag.

[20] Durbin, J. and Koopman, S.J. (2001). *Time Series Analysis by State-Space Methods*, Cambridge University Press.

[21] Elliott, R.J., Aggoun, L. and Moore, J.B. (1996). *Hidden Markov Models: Estimation and Control*, Springer-Verlag.

24
[22] Elliott, R.J., Ford, J.J. and Moore, J.B. (2000) On-line consistent estimation of hidden Markov models. Technical report, Department of Systems Engineering, Australian National University.

[23] Elliott, R.J., Ford, J.J. and Moore, J.B. (2002) On-line almost-sure parameter estimation for partially observed discrete-time linear systems with known noise characteristics. Intern. J. Adapt. Control Sig. Proc., 16, 435-453.

[24] Fearnhead, P. (2002). MCMC, sufficient statistics and particle filters. J. Comp. Graph. Statist., 11, 848–862.

[25] Ford, J.J. (1998). Adaptive hidden Markov model estimation and applications. PhD thesis, Department of Systems Engineering, Australian National University.

[26] Godsill, S.J., Doucet, A. and West, M. (2004). Monte Carlo smoothing for nonlinear time series. J. Amer. Stat. Assoc., 99, 156-168.

[27] Hernando, D., Valentino, C. and Cybenko, G. (2005) Efficient computation of the hidden Markov model entropy for a given observation sequence. IEEE Trans. Info. Theory, 51, 2681-2685.

[28] Kantas, N., Doucet, A., Singh, S.S. and Maciejowski, J.M. (2009). An overview of sequential Monte Carlo methods for parameter estimation in general state-space models. in Proceedings IFAC System Identification (SysId) Meeting.

[29] Kitagawa, G. and Sato, S. (2001). Monte Carlo smoothing and self-organising state-space model. In [19], 178–195. New York: Springer.

[30] Klaas, M., De Freitas, N. and Doucet, A. (2005). Toward practical $N^2$ Monte Carlo: The marginal particle filter. Proc. Uncertainty in Artificial Intelligence, 308-15.

[31] Le Corff, S., Fort, G. and Moulines, E. (2010) Online expectation-maximization algorithm to solve the SLAM problem. Technical report, Telecom-ParisTech.

[32] Le Gland, F. and Mevel, M. (1997) Recursive identification in hidden Markov models, Proceedings of the 36th IEEE Conference on Decision and Control, 3468-3473.

[33] Mongillo, G. and Denève, S. (2008). Online learning with hidden Markov models. Neural Computation, 20, 1706-1716.

[34] Olsson, J., Cappé, O., Douc, R. and Moulines, E. (2008). Sequential Monte Carlo smoothing with application to parameter estimation in non-linear state space models. Bernoulli, 14, 155-179.

[35] Pitt, M.K. & Shephard, N. (1999). Filtering via simulation: auxiliary particle filter. J. Am. Statist. Ass., 94, 590-9.

[36] Poyiadjis, G. (2006) Particle methods for parameter estimation in general state-space models. PhD thesis, Cambridge University.
[37] Poyiadjis, G., Doucet, A. and Singh, S.S. (2011). Particle approximations of the score and observed information matrix in state-space models with application to parameter estimation. *Biometrika*, to appear.

[38] Storvik, G. (2002). Particle filters in state space models with the presence of unknown static parameters. *IEEE. Trans. Signal Proc.*, **50**:2, 281–289.

[39] Titterington, D.M. (1984) Recursive parameter estimation using incomplete data. *J. Royal Statist. Soc.* B, 46, 257-267.

[40] Zeitouni, O. and Dembo, A. (1988). Exact filters for the estimation of the number of transitions of finite-state continuous-time Markov processes. *IEEE Trans. Inform. Theory*, **34**, 890-893.