Biased Online Parameter Inference for State-Space Models

Pierre Del Moral¹ · Ajay Jasra² · Yan Zhou²

Received: 21 October 2015 / Revised: 7 July 2016 / Accepted: 3 August 2016 / Published online: 15 August 2016 © Springer Science+Business Media New York 2016

Abstract We consider Bayesian online static parameter estimation for state-space models. This is a very important problem, but is very computationally challenging as the state-of-the-art methods that are exact, often have a computational cost that grows with the time parameter; perhaps the most successful algorithm is that of SMC² (Chopin et al., J R Stat Soc B 75: 397–426 2013). We present a version of the SMC² algorithm which has computational cost that does not grow with the time parameter. In addition, under assumptions, the algorithm is shown to provide consistent estimates of expectations w.r.t. the posterior. However, the cost to achieve this consistency can be exponential in the dimension of the parameter space; if this exponential cost is avoided, typically the algorithm is biased. The bias is investigated from a theoretical perspective and, under assumptions, we find that the bias does not accumulate as the time parameter grows. The algorithm is implemented on several Bayesian statistical models.

Keywords State-space models · Bayesian inference · Sequential Monte Carlo

Mathematics Subject Classifications (2010) Primary 82C80 · 60K35; Secondary 60F99 · 62F15

Ajay Jasra
staja@nus.edu.sg

Pierre Del Moral
pierre.del.moral@inria.fr

Yan Zhou
stazhou@nus.edu.sg

¹ Center INRIA Bordeaux Sud-Ouest & Institut de Mathématiques de Bordeaux, Université de Bordeaux I, 33405 Bordeaux Cedex, France

² Department of Statistics & Applied Probability, National University of Singapore, Singapore, 117546, Singapore
1 Introduction

We consider a state-space models, that is, a pair of discrete-time stochastic processes, \(\{X_n\}_{n\geq0}\) and \(\{Y_n\}_{n\geq1}\). The hidden process \(\{X_n\}_{n\geq0}\) is a Markov chain and the joint density for \(n\) observations of the hidden process and the observations is

\[
v_\theta(x_0) \prod_{k=1}^{n} f_\theta(x_k|x_{k-1}) g_\theta(y_k|x_k)
\]

where \(x_k \in X, k \geq 0, \) and \(y_k \in Y, k \geq 1, \) \(\theta \in \Theta \subseteq \mathbb{R}^d\) is a static parameter with prior \(\pi_\theta\). In particular, it is of interest to infer the posterior on \((\theta, x_{0:n})\), conditional upon \((y_1, \ldots, y_n)\) \((x_{0:n} = (x_0, \ldots, x_n))\) as the time parameter \(n\) grows. This problem is of interest in a wide variety of applications, including econometrics, finance and engineering; see for instance (Cappé et al. 2005).

In general, even if \(\theta\) is fixed, the posterior cannot be computed exactly and one often has to resort to numerical methods, for example by using sequential Monte Carlo (SMC) (see e.g. Doucet and Johansen 2011). SMC makes use of a collection of proposal densities and sequentially simulates from these \(N > 1\) samples, termed particles. In most scenarios it is not possible to use the distribution of interest as a proposal. Therefore, one must correct for the discrepancy between proposal and target via importance weights. In the majority of cases of practical interest, the variance of these importance weights increases with algorithmic time. This can, to some extent, be dealt with via a resampling procedure consisting of sampling with replacement from the current weighted samples and resetting them to \(1/N\). However, as is well known in the literature, due to the path degeneracy problem for particle filters, when \(\theta\) is a random variable SMC methods do not always work well; see the review of Kantas et al. (2015) for details. This has lead to a wide variety of techniques being developed, including Chopin et al. (2013), Crisan and Miguez (2014b), Fearnhead (2002), Gilks and Berzuini (2001), and Polson et al. (2008); see Kantas et al. (2015) for a full review.

The method which one might consider to be the state-of-the-art for the Bayesian online static parameter estimation for state-space models, is that in Chopin et al. (2013). This approach combines the methods of SMC samplers (Del Moral et al. 2006) and particle Markov chain Monte Carlo (PMCMC) (Andrieu et al. 2010). The method provides consistent estimates (that is, as the number of samples grows) of expectations w.r.t. the posterior on \((\theta, x_{0:n})\), as the time parameter grows. The method has been shown to work well in practice, but has one major issue; the computational cost of the application of PMCMC kernels grows as the time parameter grows; whilst the amount of times that the application of the kernel may stabilize, one will still need to apply the kernel during the algorithm (although a variety of tricks can be used to reduce the cost; see Jacob et al. 2015). We present a version of the SMC$^2$ algorithm which has computational cost that does not grow with the time parameter. In addition, under assumptions, the algorithm is shown to provide consistent estimates of expectations w.r.t. the posterior. However, the cost to achieve this consistency can be exponential in the dimension of the parameter space; if this exponential cost is avoided, typically the algorithm is biased. The bias is investigated from a theoretical perspective and, under assumptions, we find that the bias does not accumulate as the time parameter grows.

Our approach is based upon breaking up the observations into blocks of \(T\) observations; an approach used, differently, in other articles such as Chan et al. (2016), Crisan and Miguez (2014b), and Gilks and Berzuini (2001). The idea is simply to develop a ‘perfect’ SMC$^2$ algorithm which has computational cost that does not grow with the time
parameter; in practice, one cannot implement this algorithm and so one must approximate
it. Our approach simply uses the approximation of an appropriate target from the previous
block. It is remarked that if the posterior exhibits concentration/Bernstein-Von Mises prop-
ties as the time parameter grows, then alternative schemes are relevant, which could be
preferred to the ideas here; however, such properties do not always hold. See for instance the
work of Douc et al. (2011) in the context of MLE for relatively weak conditions; the prop-
ties of the MLE can impact on the concentration/Bernstein-Von Mises properties of the
posterior (see e.g. Borwanker et al. 1971).

In Section 2, SMC$^2$ is reviewed and our algorithm is presented. In Section 3 our theo-
etical results are presented, with proofs housed in the appendix. In Section 4 our simulation
results are presented. In Section 5 the article is concluded and several possible extensions
are discussed.

2 Algorithm

2.1 Notations

Let $(E, \mathcal{E})$ be a measurable space. The notation $\mathcal{B}_b(E)$ denotes the class of bounded
and measurable real-valued functions. The supremum norm is written as $\|f\|_\infty = \sup_{x \in E} |f(x)|$. $\mathcal{P}(E)$ is the set of probability measures on $(E, \mathcal{E})$. We will consider non-
negative operators $K : E \times \mathcal{E} \to \mathbb{R}_+$ such that for each $x \in E$ the mapping $A \mapsto K(x, A)$
is a finite non-negative measure on $\mathcal{E}$ and for each $A \in \mathcal{E}$ the function $x \mapsto K(x, A)$ is mea-
surable; the kernel $K$ is Markovian if $K(x, dy)$ is a probability measure for every $x \in E$.

For a finite measure $\mu$ on $(E, \mathcal{E})$, real-valued and measurable $f : E \to \mathbb{R}$

$$\mu K : A \mapsto \int K(x, A) \mu(dx) ; \quad K f : x \mapsto \int f(y) K(x, dy).$$

We also write $\mu(f) = \int f(x) \mu(dx)$. $\| \cdot \|_{tv}$ denotes the total variation
distance.

2.2 SMC$^2$

SMC$^2$ uses SMC to sample the following sequence of targets, for a fixed $N_x \geq 1$

$$\pi_n(\xi_n)d\xi_n \propto \pi_\theta(\theta)\Psi_\theta(\xi_n)\left(\prod_{k=1}^{n} \frac{1}{N_x} \sum_{i=1}^{N_x} g_\theta(y_{k}^i|x_{k}^i)\right)\prod_{k=1}^{n} \frac{1}{N_x} \sum_{i=1}^{N_x} g_\theta(y_{k-1}^i|a_{k-1}^i)$$

with $\xi_n = (\theta, x_0^{1:N_x}, a_1^{1:N_x}) \in \Theta \times \mathcal{X}^{(n+1)N_x} \times \{1, \ldots, N_x\}^n = \varnothing_n, d\xi_n$ the dominating
measure,

$$\Psi_\theta(\xi_n) = \left(\prod_{i=1}^{N_x} \nu_\theta(x_0^i)\right) \prod_{k=1}^{n} \left(\frac{1}{\sum_{j=1}^{N_x} g_\theta(y_{k-1}^i|a_{k-1}^i)} \left\{\frac{1}{N_x} \sum_{j=1}^{N_x} g_\theta(y_{k-1}^j|x_{k-1}^i)\right\} f_\theta(x_{k-1}^i|x_{k-1}^i)\right)$$
with \( g(y_0|x) = 1 \) for any \( x \in X \). Throughout, we write the normalizing constant of \( \pi_n \) as \( Z_n \). We make the following definition for notational convenience in the sequel:

\[
\beta_k(\xi_k) = \left( \prod_{i=1}^{N_x} \frac{g_\theta(y_{k-1}|x_{k-1}^i)}{\sum_{j=1}^{N_x} g_\theta(y_{k-1}|x_{k-1}^j)} f_\theta(x_k^i|x_{k-1}^i) \right) \left( \frac{1}{N_x} \sum_{i=1}^{N_x} g_\theta(y_k|x_k^i) \right) \tag{1}
\]

\[
\bar{\beta}_k(\xi_k) = \left( \prod_{i=1}^{N_x} \frac{g_\theta(y_{k-1}|x_{k-1}^i)}{\sum_{j=1}^{N_x} g_\theta(y_{k-1}|x_{k-1}^j)} f_\theta(x_k^i|x_{k-1}^i) \right).
\tag{2}
\]

We provide a Feynman-Kac representation of the SMC\(^2\) algorithm which facilitates theoretical analysis and will assist our subsequent notations. Set

\[
\eta_0(\xi_1) = \pi_\theta(\theta) \left( \prod_{i=1}^{N_x} v_\theta(x_0^i) \right) \left( \prod_{i=1}^{N_x} \frac{g_\theta(y_0|x_0^i)}{\sum_{j=1}^{N_x} g_\theta(y_0|x_0^j)} f_\theta(x_1^i|x_0^i) \right)
\]

and for any \( k \geq 1 \), \( M_k : E_k \to \mathcal{P}(E_{k+1}) \),

\[
M_k(\xi_k, d\xi_{k+1}) = \tilde{M}_k(\xi_k, d\tilde{\xi}_{k+1}) \left( \prod_{i=1}^{N_x} \frac{g_\theta(y_k|x_k^i)}{\sum_{j=1}^{N_x} g_\theta(y_k|x_k^j)} f_\theta(x_{k+1}^i|x_k^i) \right) d\chi_{k+1}^{1:N_x}
\]

where \( \tilde{\xi}_{k+1} = (\tilde{\xi}_k, a_k^{1:N_x}, x_{k+1}^{1:N_x}) \), \( \tilde{M}_k \) is \( \pi_\theta \)-invariant (e.g. a particle marginal Metropolis-Hastings (PMMH) kernel Andrieu et al. 2010). Then we set

\[
G_k(\xi_{k+1}) = \frac{1}{N_x} \sum_{i=1}^{N_x} g_\theta(y_{k+1}|x_{k+1}^i).
\]

and for \( \varphi \in B_\theta(E_{n+1}) \)

\[
\gamma_n(\varphi) = \int \varphi(\xi_{n+1}) \prod_{p=0}^{n-1} G_p(\xi_{k+1}) \eta_0(\xi_1) \prod_{k=1}^{n} M_k(\xi_k, d\xi_{k+1}) d\xi_1.
\]

The SMC algorithm which approximates the \( n \)-time marginal

\[
\eta_n(\varphi) = \frac{\gamma_n(\varphi)}{\gamma_n(1)}
\]

will have joint law:

\[
\prod_{i=1}^{N} \eta_0(\xi_{i}^1)d\xi_{i}^1 \prod_{k=1}^{n} \left( \prod_{i=1}^{N} \Phi_k(\eta_{k-1}^N)(d\xi_{i,k+1}^k) \right)
\tag{3}
\]

where \( \Phi_k(\mu)(dx) = \mu(G_{k-1}M_k(dx)) / \mu(G_{k-1}) \) is the usual selection-mutation operator and \( n_{k-1}^N \) is the empirical measure of the particles. It is easily shown that by approximating \( \eta_n \) one can approximate the posterior on \( \theta \) and \( x_{0:n} \) given \( y_{1:n} \); see Chopin et al. (2013) and Del Moral (2013). We note that, typically one will apply the kernel \( M_k \) when dynamic resampling is performed; however the form of the algorithm is simple to describe with resampling at each time step.

### 2.3 Perfect Algorithm

One issue with the above algorithm is that whenever the kernel \( \tilde{M}_k \) is applied, one must sample trajectories of the hidden states which grow with the time parameter. Thus, even
if resampling is dynamically performed, leading to an application of $\hat{M}_k$, the cost of the algorithm will increase with the time parameter. So one can say whilst SMC$^2$ is a very powerful algorithm, it is not an online algorithm. Here we present an ‘ideal’ or perfect version of the algorithm that has computational cost which does not grow with time, but cannot be implemented in general. This algorithm will provide the basis for our biased algorithm in the next section.

To begin, we set $B \in \mathbb{Z}^+$ which in principle can grow and $T \in \mathbb{Z}^+$ which is fixed. The parameter $T$ will represent the maximum length of the trajectory of the hidden state, which one wants to sample. We define the following target probabilities:

$$\pi_{n,T}(\xi_n)d\xi_n \propto \pi_\theta(\theta) \left( \prod_{i=1}^{N_x} \nu_\theta(x_0^i) \right) \prod_{k=1}^n \beta_k(\xi_k)d\xi_n \quad n \in \{0, \ldots, T\} \quad (4)$$

and set $\bar{\xi}_n = (\theta, x_1^{(b-1)T+1:n}, a_1^{(b-1)T+2:n})$

$$\pi_{n,bT}(\bar{\xi}_n)d\bar{\xi}_n \propto \zeta_{b,T}(\theta) \left( \prod_{i=1}^{N_x} \zeta_{b,T}(x_0^i) \right) \times$$

$$\left( \frac{1}{N_x} \sum_{i=1}^{N_x} g_\theta(y_{(b-1)T+1}^i|x_{(b-1)T+1}^i) \right) \prod_{k=(b-1)T+2}^n \beta_k(\xi_k)d\bar{\xi}_n \quad (5)$$

$n \in \{(b-1)T + 1, \ldots, bT\}, b \in \{2, \ldots, B\}$

where $\beta_k$ is as Eq. 1 and

$$\zeta_{b,T}(\theta, x_{(b-1)T+1}) \propto \pi_\theta(\theta) \int f_\theta(x_{(b-1)T+1}|x_{(b-1)T})v_\theta(x_0) \times$$

$$\prod_{k=1}^{(b-1)T} f_\theta(x_k|x_{k-1})g_\theta(y_k|x_k)dX_0(1:bT)$$

with $\zeta_{b,T}(\theta) = \int \zeta_{b,T}(x_{(b-1)T+1}, \theta)dx_{(b-1)T+1}$ and $\zeta_{b,T}(x_{(b-1)T+1}|\theta)$ is the associated conditional. If one can approximate the targets above one can approximate the posterior on $\theta$ and $x_0^n$ given $y_{1:n}$. As we will see below, our algorithm to achieve this cannot be implemented in practice, but has the benefit that the computational cost per-time step cannot grow beyond a given bound.

2.3.1 Algorithm

For $n \in \{0, \ldots, T\}$ one can run the SMC$^2$ algorithm as in Section 2.2. That is to run the algorithm with law (3) until time $T-1$. We will add a final time step that will resample the $N$ particles according to $G_{T-1}$. That is, defining $\hat{M}_T$ as the Dirac measure, we sample from

$$\prod_{i=1}^N \eta_0(\xi_1^i)d\xi_1^i \prod_{k=1}^T \left( \prod_{i=1}^N \Phi_k(\eta^{N_i}_{k-1})(d\xi_k^i) \right) .$$

For the subsequent blocks, we make the following definitions. Let $b \in \{2, \ldots, B\}$, $\bar{\xi}_{(b-1)T+1} = (\theta, x_1^{(b-1)T+1}) \in \Theta \times X^{N_x} = \hat{\xi}_{(b-1)T+1}$, $\bar{\xi}_n = (\theta, x_{(b-1)T+1:n+1}, a_1^{(b-1)T+2:n+1}) \in \Theta \times X^{n-(b-1)T+1}N_x \times \{1 \ldots, N_x\}^{(n-(b-1)T)N_x} = \hat{\xi}_n$. 

© Springer
n \in \{(b-1)T+1, \ldots, bT-1\}, and \(\tilde{\xi}_{bT} = (\theta, x_{(b-1)T+1:bT}^1, a_{(b-1)T+2:bT}^1) \in \Theta \times X^{T\cdot N_x} \times \{1, \ldots, N_x\}^{(T-1)N_x} = \tilde{E}_{bT}\). Now let

\[
\tilde{G}_n(\tilde{\xi}_n) = \frac{1}{N_x} \sum_{i=1}^{N_x} g_0(y_n|x_n^i) \quad n \in \{(b-1)T + 1, \ldots, bT\}.
\]

Set

\[
\tilde{\eta}_{(b-1)T+1}(\tilde{\xi}_{(b-1)T+1}) = \zeta_{bT}(\theta)\left(\prod_{i=1}^{N_x} \zeta_{bT}(x_n^i_{(b-1)T+1+1} | \theta)\right).
\]

Define \(\tilde{M}_{n,bT,\pi_{n-1},bT}\) as a Markov kernel of invariant density \(\pi_{n-1},bT, n \in \{(b-1)T + 2, \ldots, bT-1\}\) such as a PMMH kernel. Then define

\[
\tilde{M}_{n,bT,\pi_{n-1},bT}(\tilde{\xi}_{n-1},d\tilde{\xi}_n) = \tilde{M}_{n,bT,\pi_{n-1},bT}(\tilde{\xi}_{n-1},d\tilde{\xi}_n\tilde{\eta}_{n}^1)\tilde{\eta}_{n+1}(\tilde{\xi}_{n+1})d(x_{n+n+1}^1)
\]

where \(\tilde{\xi}_n = (\tilde{\xi}_{n-1}^1, a_{n+n+1}^1), (a_{n+n+1}^1)'\) and \(\tilde{\eta}\) is as Eq. 2. Set \(\tilde{M}_{bT,bT,\pi_{bT-1},bT}\) as a Dirac mass. Finally, for \(\mu \in \mathcal{P}(\tilde{E}_{n-1}), n \in \{(b-1)T + 2, \ldots, bT\}\) and probability density \(\psi\) on \(\tilde{E}_{n-1}\) define

\[
\tilde{\Phi}_{n,bT,\psi}(\mu)(d\tilde{\xi}_n) := \frac{\mu(\tilde{G}_{n-1}\tilde{M}_{n,bT,\pi_{bT},\psi}(d\tilde{\xi}_n))}{\mu(\tilde{G}_{n-1})}.
\]

Then the perfect algorithm has joint law for \(b \in \{2, \ldots, B\}, n \in \{(b-1)T + 1, \ldots, bT\}\)

\[
\prod_{i=1}^{N} \tilde{\eta}_{(b-1)T+1}(\tilde{\xi}_{(b-1)T+1}^i) \prod_{k=(b-1)T+2}^{n} \tilde{\Phi}_{k,bT,\pi_{k-1},bT}(\tilde{\eta}_{k-1,bT}^N)(d\tilde{\xi}_k^i)
\]

where \(\tilde{\eta}_{k-1,bT}^N\) is the empirical measure of the particles at time \(k - 1\).

The algorithm may be summarized in words as follows:

- Run an SMC\(^2\) algorithm up-to time \(T\).
- Then, we will run a SMC\(^2\) algorithm for the data between \(T+1\) and \(2T\). The algorithm is only run for times \(T+1\) up to \(2T\) where one substitutes the exact marginal on \(\theta|y_{1:T}\) (for the prior on \(\theta\)) and the exact conditional \(x_{T+1}|\theta, y_{1:T}\) for the intial density of the \(N_x\) particles.
- Subsequent blocks are similar. E.g. in the third block we will run a SMC\(^2\) algorithm for the data between \(2T+1\) and \(3T\), etc.

### 2.3.2 Remarks

Set, for \(b \in \{2, \ldots, B\}, n \in \{(b-1)T + 2, \ldots, bT\}\)

\[
\tilde{\gamma}_{n,bT}(d\tilde{\xi}_n) = \int_{\tilde{E}_{(b-1)T+1} \times \cdots \times \tilde{E}_{n-1}} \prod_{k=(b-1)T+1}^{n-1} \tilde{G}_k(\tilde{\xi}_k) \tilde{\eta}_{(b-1)T+1}(d\tilde{\xi}_{(b-1)T+1}) \times \prod_{k=(b-1)T+1}^{n} \tilde{M}_{k,bT,\pi_{k-1},bT}(\tilde{\xi}_{k-1},d\tilde{\xi}_k)
\]
and \( \tilde{\eta}_{n,bT}(d\tilde{\xi}_n) = \tilde{\gamma}_{n,bT}(d\tilde{\xi}_n)/\tilde{\gamma}_{n,bT}(1) \), then we note that for \( \varphi : \tilde{E}_{n-1} \to \mathbb{R} \), \( \pi_{n,bT} \) is-integrable

\[
\tilde{\eta}_{n,bT}(\tilde{G}_n\varphi) = \int_{\tilde{E}_{n-1}} \varphi(\theta, x_{(b-1)T+1:n}, a_{(b-1)T+2:n})
\]

\[
\pi_{n,bT}(\theta, x_{(b-1)T+1:n}, a_{(b-1)T+2:n})d(\theta, x_{(b-1)T+1:n})
\]

so that one can estimate expectations w.r.t. \( \pi_{n,bT} \) via

\[
\frac{\tilde{\eta}_{n,bT}(\tilde{G}_n\varphi)}{\tilde{\eta}_{n,bT}(\tilde{G}_n)}.
\]

If \( \tilde{M}_{n,bT,\pi_{n-1,bT}} \) is a PMMH kernel with proposal:

\[
q(\theta, \theta') \left( \prod_{i=1}^{N_x} \zeta_{b,T}(x_{(b-1)T+1}^{i,\prime} | \theta') \right) \prod_{k=(b-1)T+2}^{n-1} \tilde{\beta}(\xi_k^i)
\]

where \( \prime \) denotes proposed value and \( q \) is a Markov density on \( \Theta \) then the acceptance probability is:

\[
\min \left\{ 1, \frac{\prod_{k=(b-1)T+1}^{n-1} N_x^i \sum_{i=1}^{N_x} \varphi(\theta, x_{(b-1)T+1}^{i,\prime}) \zeta_{b,T}(\theta')q(\theta', \theta) \zeta_{b,T}(\theta)q(\theta, \theta')} {\prod_{k=(b-1)T+1}^{n-1} N_x^i \sum_{i=1}^{N_x} \varphi(\theta, x_{(b-1)T+1}^{i}) \zeta_{b,T}(\theta)q(\theta, \theta')} \right\}.
\]

### 2.4 Approximate Algorithm

The problem with the previous algorithm is that one can seldom evaluate \( \zeta_{b,T}(\theta, x_{(b-1)T+1}) \) nor sample from \( \zeta_{b,T}(\theta) \) nor \( \zeta_{b,T}(x_{(b-1)T+1} | \theta) \). We introduce the following algorithm, which will sample from the following targets. For the first block, one can run the SMC\(^2\) algorithm to target the sequence (4). At the subsequent blocks, one is unable to evaluate the target, nor sample from the proposals. We propose the following approximate targets to replace (5):

\[
\tilde{\pi}_{n,bT}(\theta, x_{(b-1)T+1:n}, a_{(b-1)T+2:n}) \propto \frac{1}{N} \sum_{j=1}^{N} K_N(\theta - \theta^j) \left( \prod_{i=1}^{N_x} \sum_{j=1}^{N} K_N(\theta - \theta^j) \right) \times
\]

\[
f_{\theta}(x_{(b-1)T+1}^{i} | x_{(b-1)T}^{i}) \left( \frac{1}{N_x} \sum_{i=1}^{N_x} g_{\theta}(y_{(b-1)T+1}^{i} | x_{(b-1)T}^{i}) \right) \prod_{k=(b-1)T+2}^{n} \beta_k(\xi_k) \]

with \( n \in \{(b-1)T + 1, \ldots, bT\}, b \in \{2, \ldots, B\}, \theta^{1:N}, x_{(b-1)T}^{1:N} \) samples from the algorithm at the previous block, which we shall describe how to obtain and \( K_N(\theta - \theta') \) a kernel density whose bandwidth may depend on \( N \). Here the term \( \frac{1}{N} \sum_{j=1}^{N} K_N(\theta - \theta^j) \) approximates \( \zeta_{b,T}(\theta) \) and the term

\[
\sum_{j=1}^{N} \frac{K_N(\theta - \theta^j)}{\sum_{s=1}^{N} K_N(\theta - \theta^s)} f_{\theta}(x_{(b-1)T+1}^{i} | x_{(b-1)T}^{i})
\]

is an approximation of \( \zeta_{b,T}(x_{(b-1)T+1} | \theta) \). The rationale for using a kernel density is allow a support for \( \theta \) that is exactly \( \Theta \) and not constrained to the current set of particles (i.e. using a mixture of Dirac measures concentrated on the current particle set).
At the end of a block of the algorithm, we just take $\theta^{1:N_x}$ and $x_i^{1:N_x}$ as the samples we have obtained, taking the first of the $N_x$-tuples of $x_i^{1:N_x}$ (as the samples are exchangeable). Using standard SMC theory, which we will expand upon, one can prove that if the bandwidth of $K$ appropriately depends on $N$ that at least $\hat{\pi}_{n,2T}(\theta, x_i^{1:N_x}, a_T^{+1}; a_{2n})$ will converge almost surely (in an appropriate sense) to $\pi_{n,2T}(\theta, x_i^{1:N_x}, a_T^{+1}; a_{2n})$ as $N$ grows, hence providing the justification of the approximation introduced.

Set

$$\hat{\eta}_{(b-1)T+1}(\xi(b-1)T+1) = \frac{1}{N} \sum_{j=1}^{N} K_N(\theta - \theta^j) \left( \prod_{i=1}^{N_x} \frac{1}{N} \sum_{j=1}^{N} K_N(\theta - \theta^j) \times f_{\theta}(x_i^{(b-1)T+1} | x_j^{(b-1)T}) \right).$$

Then the approximate algorithm has joint law for $b \in \{2, \ldots, B\}$, $n \in \{(b-1)T + 1, \ldots, bT\}$

$$\prod_{i=1}^{N} \hat{\eta}_{(b-1)T+1}(\xi_i^{(b-1)T+1}) \prod_{k=(b-1)T+2}^{n} \prod_{i=1}^{N} \hat{\phi}_{k,bT,\hat{\pi}_{k-1,bT}}(\hat{\eta}_{k-1,bT}) (d\xi_k^{(b-1)T+1})$$

where $\hat{\eta}_{k-1,bT}$ is the empirical measure of the particles at time $k - 1$. An estimate of the form (6) can be used to estimate the targets.

The algorithm may be summarized in words as follows:

- Run an SMC$^2$ algorithm up-to time $T$.
- Then, we will run a SMC$^2$ algorithm for the data between $T + 1$ and $2T$. The algorithm is only run for times $T + 1$ up to $2T$ where one substitutes an approximation of the true marginal on $\theta | y_1:T$ (for the prior on $\theta$) and an approximation of the exact conditional $x_T | y_1:T$ for the initial density of the $N_x$-particles.
- Subsequent blocks are similar. E.g. in the third block we will run a SMC$^2$ algorithm for the data between $2T + 1$ and $3T$, etc.

Note that the cost of the algorithm is $O(N^2)$. For instance, in a PMMH kernel $\tilde{M}_{n,bT,\tilde{\pi}_{n-1,bT}}$ with proposal:

$$q(\theta, \theta') \left( \prod_{i=1}^{N_x} \sum_{j=1}^{N} \sum_{s=1}^{N_x} K_N(\theta' - \theta^j) f_{\theta'}(x_i^{(b-1)T+1} | x_j^{(b-1)T}) \right) \prod_{k=(b-1)T+2}^{n-1} \tilde{\beta}(\xi_k^{(b-1)T+1})$$

the acceptance probability is

$$\min \left\{1, \frac{1}{N_x} \sum_{i=1}^{N_x} g_{\theta'}(y_k | x_i^{(b-1)T+1}) \sum_{j=1}^{N} K_N(\theta - \theta^j) q(\theta', \theta) \right\}.$$

Whilst the evaluation of $\sum_{j=1}^{N} K_N(\theta - \theta^j)$ can be circumvented by using an auxiliary variable trick, the simulation of $\prod_{i=1}^{N_x} \sum_{j=1}^{N} \sum_{s=1}^{N_x} K_N(\theta' - \theta^j) f_{\theta'}(x_i^{(b-1)T+1} | x_j^{(b-1)T})$ is $O(N)$. It should be noted that the use of PMMH could be replaced, for instance, by the correlated pseudo marginal (Deligiannidis et al. 2015) algorithm and other related MCMC kernels.
2.5 Related Simulation Methods and Alternatives

Similar, but different, ideas have appeared in several articles including (Andrieu et al. 2009; Centanni and Minozzo 2006; Crisan and Miguez 2014b) these ideas are considered in the context of hidden Markov models and partially observed point processes respectively. The key differences of our work to Centanni and Minozzo (2006) and Crisan and Miguez (2014b) ((Andrieu et al. 2009) is for maximum likelihood estimation (MLE)) are as follows.

In the context of Centanni and Minozzo (2006) we do not use a type of ‘sequential MCMC’, in that our approach can be used explicitly for online Bayesian parameter estimation. The approach of Crisan and Miguez (2014b) is less general, where the particles are not updated with PMCMC, and one has a block length of 1.

Alternatives to the approach outlined above are; (i) a form of sequential PMCMC in the spirit of Centanni and Minozzo (2006) or (ii) to use an over-lapping or sliding window. For (i), one expects that the cost is higher than the above algorithm, and online (filtered) estimates are not available. For (ii) the cost may be higher, but, in simulation studies, we did not find any obvious improvement in practice. We also remark that our approximate algorithm uses kernel density estimation, but, in principle, any approximation scheme could be used. In practice, the algorithm is implemented with dynamic resampling according to the effective sample size.

3 Theoretical Results

Throughout, it is supposed that for any \( n \geq 1 \sup_{\theta, x} \phi_{\theta}(y_n | x) < +\infty \).

3.1 Consistency

We will now show that, under a specific choice of the bandwidth \( h \) and under some mathematical assumptions, the algorithm just presented is consistent. We set \( K_h(\theta) = h^{-d} K(h^{-1} \theta) \); it is supposed that for any fixed \( \theta \), \( \lim_{h \to 0} K_h(\theta) = 0 \). For simplicity we denote \( \eta_n(x) \) (resp. \( E_n \)), \( n \in \{0, \ldots, T\} \) as \( \tilde{\eta}_{n,T}(x) \) (resp. \( \tilde{E}_n \)).

(A1) We have

\[
\begin{align*}
&- \int_\Theta K(\theta) \, d\theta = 1, K(\theta) \geq 0 \forall \theta \in \Theta \\
&- \int \|\theta\|^2 K(\theta) \, d\theta < +\infty \\
&- K \text{ is bounded and continuous in the } L_1 \text{ metric.}
\end{align*}
\]

(A2) For each \( n, b, b \in \{1, \ldots, B\} \), \( n \in \{(b-1)T+1, \ldots, bT\} \) there exist a \( L_{n,k} > 0 \) such that for every \( \theta, \theta' \in \Theta \)

\[
|\tilde{\eta}_{n,bT}(\theta) - \tilde{\eta}_{n,bT}(\theta')| \leq L_{n,k} \|\theta - \theta'\|
\]

where \( \tilde{\eta}_{n,bT}(\theta) \) is the marginal on \( \theta \).

(A3) We have \( h = N^{-\frac{1}{2(d+1)}} \), in addition there exist a \( C > 0 \) such that for every \( N \sup_{\theta \in \Theta} |K_N(\theta)| \leq C \).

(A4) For any \( b \in \{2, \ldots, B\} \), \( n \in \{(b-1)T+1, \ldots, bT\} \), \( \varphi \in \mathcal{B}(\tilde{E}_n) \) the function \( \eta \mapsto \tilde{M}_{n,bT,\eta}(\varphi)(x) \) is continuous (in the \( L_1 \)-metric) at \( \eta = \pi_{n-1,bT} \) uniformly in \( x \in \tilde{E}_{n-1} \).
The first three assumptions are from Crisan and Miguez (2014a) to allow convergence of the SMC estimate of the kernel density and the final assumption is from Beskos et al. (2016). We set

\[ \hat{\eta}^N_{(b-1)T,(b-1)T}(K_N(\theta)f_\theta(x_{(b-1)T+1}|x_{(b-1)T})). \]

→_P denotes convergence in probability as \( N \to \infty \). We now have the following result whose proof is in Appendix A.

**Theorem 3.1** Assume (A1-4). Then for each \( n, b, b \in \{2, \ldots, B\}, n \in \{(b-1)T + 1, \ldots, bT\} \) and \( \varphi \in \mathcal{B}_b(\hat{E}_n) \) we have

\[ \hat{\eta}^N_{n,bT}(\varphi) \to \hat{n}_{n,bT}(\varphi) \]

and in particular for every \( \theta \in \Theta, x_{(b-1)T+1} \in X \)

\[ \hat{\eta}^N_{(b-1)T,(b-1)T}(K_N(\theta)f_\theta(x_{(b-1)T+1}|x_{(b-1)T})) \to \hat{\eta}_{n,bT}(\varphi). \]

**Remark 3.1** The result here is essentially qualitative. In order to obtain consistency, one must adopt an exponential effort in \( d \) and so one would only run (with the choice of \( h \) as in (A3)) over the exact algorithm if the time horizon is long and \( d \) is small. In practice one will decouple \( h \) and \( N \), leading to estimates which are biased, even if \( N \to \infty \). We will now study the bias.

### 3.2 Bias

We now investigate the asymptotic (in \( N \)) bias of the approach. It should be noted that \( h \) is now fixed. We make the following hypothesis:

**(A5)** There exist a \( \delta \in [1, \infty) \) such that for any \( n \geq T + 1 \):

\[ \sup_{(x,y) \in \hat{E}_n} \frac{\hat{G}_n(x)}{\hat{G}_n(y)} \leq \delta. \]

There exist \( \epsilon \in (0, 1) \) and for any \( b \in \{2, \ldots, B\}, n \in \{(b-1)T + 2, \ldots, bT-1\} \) there exist a \( \nu \in \mathcal{P}(\hat{E}_n) \) and any probability density \( \psi \) on \( \hat{E}_{n-1} \), \( (x, y) \in \hat{E}_{n-1}^2 \)

\[ \bar{M}_{n,bT,\psi}(x, \cdot) \geq \epsilon \left( \bar{M}_{n,bT,\psi}(y, \cdot) \wedge \nu(\cdot) \right). \]

We write the \( N \)-limiting version of Eq. 7 as \( \hat{\eta}_{n,bT,\psi} \) (which one can easily prove exists, for any \( b \in \{2, \ldots, B\} \)). For a bounded \( \varphi : \mathbb{E}_{bT} \to \mathbb{R} \) we define the bias as

\[ B(bT, \varphi) = \left| \mathbb{E}_{\pi_{bT,\psi}^n} - \mathbb{E}_{\pi_{bT,\psi}^n} \right| \varphi( X_{bT}^{1:N_b}, d_{bT}^{1:N_b}, \theta) \].

We have the following result whose proof is in Appendix B.

**Theorem 3.2** Assume (A5). Then there exist a \( C < +\infty, \rho_1, \rho_2 \in (0, 1) \) such that for any \( b \in \{2, \ldots, B\} \):

\[ B(bT, \varphi) \leq C \| \varphi \|_\infty (1 - \rho_1^{T-1} + \rho_2^{T-1}) \].

**Remark 3.2** The result indicates that some of the bias can fall at a geometric rate as \( T \) grows. Note, that the bias cannot disappear once, one block is wrong (which is by assumption in
the statement of the theorem). The result also provides the reassuring point that bias’ do not accumulate as the number of blocks grow, albeit under strong assumptions.

4 Simulations

4.1 Gaussian Linear Model

4.1.1 Model and Algorithm Setting

We consider the following Gaussian model,

\[
\begin{align*}
\nu_\theta(x_0) &= \phi(x_0; 0, \tau_0^{-1}) \\
f_\theta(x_k | x_{k-1}) &= \phi(x_k; x_{k-1}, \tau^{-1}) \\
g_\theta(y_k | x_k) &= \phi(y_k; x_k, \lambda^{-1})
\end{align*}
\]

![Fig. 1](image-url) Average of estimates of state $X_k$ for the Gaussian linear model
where $\phi(x; \mu, \sigma^2)$ is the density function of a Normal distribution with mean $\mu$ and variance $\sigma^2$.

The data is simulated from $\tau_0 = \tau = \lambda = 1$, with 10,000 time steps. The algorithms are implemented with $\tau_0$ fixed to 1 and the other two being estimated. That is, the parameter is $\theta = (\tau, \lambda)$.

We consider three algorithms. First, due to the simple structure of the model, we can obtain exact evaluation of $p(y_k | y_{1:k-1})$ through a Kalman filter. We will replace the particle filter in the SMC$^2$ algorithm with the Kalman filter; this results in a regular SMC algorithm. This is used to provide accurate unbiased estimators when comparing algorithms. The second is the SMC$^2$ algorithm and the third is the proposed new algorithm, SMC$^2$FW. The PMCMC step in each algorithm is constructed as a two-block Metropolis random walk, with a Normal kernel on the logarithm scale.

With the Kalman filter, the computational cost of $p(y_k | y_{1:k-1})$ is negligible and thus we use a large number of $\theta$-particles, $N = 10000$. This allows us to obtain unbiased posterior estimators with very small variance. As a result, this provides a good baseline for the comparison of the other two algorithms.

![Fig. 2](image)

**Fig. 2** Average of estimates of parameter $\tau$ (transition precision) for the Gaussian linear model
For the SMC$^2$ algorithm, we use $N = 1000$ particles for the SMC algorithm and $N_x = 1000$ particles. For simplicity, the value of $N_x$ is fixed through the time line.

For the SMC$^2$FW algorithm, we set $N = 10,000$ and $N_x = 1000$. As we will see later, despite the increased number of the $\theta$-particles, the computational cost is still significantly lower than SMC$^2$, in addition to be upper bounded per time step. The kernel density estimate (KDE) is bivariate Normal on the logarithm scale, with covariance matrix taken a diagonal form, $\Sigma = hI_2$ where $I_2$ is the identity matrix of rank two. We consider $h = 0.01, 0.05$ and $0.25$. Three widths of each fixed window $T = 125, 500$ and $1000$ are also considered.

4.1.2 Results

In Figs. 1, 2 and 3 we show the average of estimates over 30 simulations, given different bandwidth $h$ and window width $T$. The first 125 time steps are cutoff from the graphs, during which time the SMC$^2$FW algorithm is exactly the same as the SMC$^2$ algorithm.

![Fig. 3 Average of estimates of parameter $\lambda$ (observation precision) for the Gaussian linear model](image-url)
Algorithms labeled with “Kalman” means a Kalman filter is used in place of a particle filter and thus exact values of $p(y_k | y_{1:k-1})$ are calculated instead of approximations. In this case, the results show the behavior of the $\text{SMC}^2\text{FW}$ algorithm when $N_k \to \infty$. However, in this particular example, errors introduced by the particle filter approximation is minimal as shown in the graphs.

It is as expected that the longer the window width, the better the $\text{SMC}^2\text{FW}$ algorithms perform. The choice of the bandwidth $h$ has a more dramatic effect on the performance. With $h = 0.01$ the algorithms give almost exact results even for $T = 125$. On the other hand, with $h = 0.25$, the errors are numerous. For the state $X_k$, neither the bandwidth nor the window width affect the results in any observable way. For the parameters, $h = 0.01$ and $T = 500$ provides the best results.

In Fig. 4 the MSE of the two algorithms are shown, using the results obtained with a SMC algorithm using Kalman filters ($\text{SMC}^2$ (Kalman) in previous figures) as an unbiased, accurate estimate of the true posterior means. It can be seen that for the state $X_k$, the results

![Figure 4](image_url)

*Fig. 4* MSE of estimates of parameters and states for the Gaussian linear model. The $\text{SMC}^2\text{FW}$ algorithm uses $h = 0.01$ and $l = 500$
are mixed. For the $\tau$ parameter the $\text{SMC}^2$ algorithm has a smaller MSE while the opposite is true for the $\lambda$ parameter. It shall be noted that, the $\text{SMC}^2$FW algorithm use $N = 10,000$ while the $\text{SMC}^2$ algorithm only use $N = 1000$. However, despite the large difference of the numbers of $\theta$-particles, the $\text{SMC}^2$FW algorithm is still significantly more computationally cost efficient. It took about thirty minutes for a single run of the $\text{SMC}^2$FW algorithm under this setting while it took about five hours for the $\text{SMC}^2$ algorithm. More importantly, the cost of $\text{SMC}^2$FW is bounded per time step, and thus it is possible to obtain better results than $\text{SMC}^2$ for all parameters while having a bounded, smaller cost in the long run.

4.2 Lévy-Driven Stochastic Model

4.2.1 Model and Algorithm Setting

We consider the Lévy-driven stochastic volatility model, applied to 1,000 recent S&P 500 data from September 17, 2010 to September 8, 2014. The data are obtained as logarithm return of the daily adjusted close price, and then normalized to unity variance. The data is plotted in Fig. 5. There are considerably much larger volatility at the beginning of the series. It becomes much more stable in the middle and slightly larger at the end.

The model we employed to analyze this data is the same in Andrieu et al. (2010) and we use the same notations and formulation as in that paper. The model has four parameters $\theta = (\kappa, \delta, \gamma, \lambda)$ and a two-dimensional state $X_n = (\sigma^2(n\Delta), z(n\Delta))$ where $\Delta$ is the time interval and in this example it set to constant 1.

We consider two algorithms. First, the $\text{SMC}^2$ algorithm with $N = 1000$ and $N_x = 1000$. Second, the $\text{SMC}^2$FW algorithm with the same number of particles. The same Normal KDE approximation is used as in the last example. Various bandwidths of the kernel were considered, and $h = 0.01$ is chosen. The latter algorithm uses a window width $T = 200$. We found that further increasing the window width does not improve results in a significant manner.

The algorithms again use Normal kernels on logarithm scales for the PMMH proposals. Using preliminary results, we believe that $\kappa$ and $\delta$ are strongly correlated and they are

![Fig. 5 S&P 500 daily log-return data](image-url)
updated in one block of PMCMC move while $\gamma$ and $\lambda$ are updated individually in their own block. The proposal scales are calibrated on-line using the moments estimates from the sampler.

4.2.2 Results

In Fig. 6 we show the average of estimates over 20 simulations. Both algorithms give similar results.

One feature that was mentioned in Section 3, but not shown clearly in the early simple Gaussian linear example, is that the bias of the online algorithms does not accumulate over time. For instance, consider the $\sigma^2$ state, though non-trivial errors can be observed for both online algorithms in the early time steps, they were not carried on to later times.

In addition to the estimates, prediction of the variance (square of volatility), $\sigma^2(n \Delta)$ is also calculated as $\mathbb{E}[X_n|y_{1:n-1}] = \mathbb{E}[\mathbb{E}[X_n|X_{n-1}]|y_{1:n-1}]$. Since the transition density is not of a closed form, its expectation is estimated with 100 samples of $X_n$ generated for each
value of $X_{n-1}$ in the particle system. The outer expectation is approximated with the particle system at time $n - 1$. The results for the three algorithms are plotted in Fig. 7 against the squared log-returns.

4.2.3 Summary

Through two examples, it is shown that with some careful choice of the KDE bandwidth, it is possible to obtain results competitive to the SMC$^2$ algorithm with only a fraction of computational cost. In the Lévy-driven stochastic model example, the theoretical feature that the bias does not accumulate over time is demonstrated.

5 Summary

In this article we have presented a method for Bayesian online static parameter estimation for state-space models. Our method is such that the computational cost does not grow with the time parameter and moreover, the algorithm can be shown to be consistent, although, the cost is then exponential in the dimension of the static parameter. We have additionally shown that the asymptotic bias, under strong assumptions, does not grow with time.

There are several avenues for future work. First is the use of alternative approximation schemes in our algorithm; we have relied on kernel density estimation, but there are other schemes which could be used. Second, in our work, we have investigated the asymptotic bias. However, as is clear in the proofs, then the consecutive blocks are independent and one thus expects that the study of the finite sample bias is significantly more challenging; an investigation of this is warranted.

Acknowledgments AJ was supported by a Singapore Ministry of Education Academic Research Fund Tier 1 grant (R-155-000-156-112). We thank Alex Beskos for many useful conversations on this work. We also thank the referee for comments which have greatly improved the article.
Appendix A: Technical Results: Consistency

Proof of Theorem 3.1 For $n \in \{0, \ldots, T\}$ the result follows by standard theory; see Crisan and Miguez (2014a), Del Moral (2004), and Del Moral (2013). We first consider $\tilde{\eta}_{T,T}^N(K_N(\theta)f_{\theta}(x_{T+1}^T))$. Denote by $\mathcal{F}_T$ the $\sigma-$algebra generated by the particle system up to time $T$ and expectations w.r.t. the law of the simulated algorithm as $\mathbb{E}$. Then we have

$$
\tilde{\eta}_{T,T}^N(K_N(\theta)f_{\theta}(x_{T+1}^T)) - \zeta(\theta, x_{T+1}) = \tilde{\eta}_{T,T}^N(K_N(\theta)f_{\theta}(x_{T+1}^T)) - \mathbb{E}[\tilde{\eta}_{T,T}^N(K_N(\theta)f_{\theta}(x_{T+1}^T))|\mathcal{F}_T] + \mathbb{E}[\tilde{\eta}_{T,T}^N(K_N(\theta)f_{\theta}(x_{T+1}^T))|\mathcal{F}_T] - \zeta(\theta, x_{T+1})
$$

where we have dropped the subscripts on $\zeta$. For the first term on the R.H.S. we have

$$
\mathbb{E}[\tilde{\eta}_{T,T}^N(K_N(\theta)f_{\theta}(x_{T+1}^T)) - \mathbb{E}[\tilde{\eta}_{T,T}^N(K_N(\theta)f_{\theta}(x_{T+1}^T))|\mathcal{F}_T]] \leq \frac{C'}{\sqrt{N}}
$$

where $C'$ depends on $C$ in (A3); thus $\tilde{\eta}_{T,T}^N(K_N(\theta)f_{\theta}(x_{T+1}^T)) - \mathbb{E}[\tilde{\eta}_{T,T}^N(K_N(\theta)f_{\theta}(x_{T+1}^T))|\mathcal{F}_T]$ converges to zero in probability.

Now, for $\mathbb{E}[\tilde{\eta}_{T,T}^N(K_N(\theta)f_{\theta}(x_{T+1}^T))|\mathcal{F}_T] - \zeta(\theta, x_{T+1})$

For $\tilde{\eta}_{T+1,2T}^N(\varphi)$, this converges in probability to $\tilde{\eta}_{T+1,2T}(\varphi)$ by centering the conditional expectation (given $\mathcal{F}_{T+1}$) and applying the (conditional) Marcinkiewicz-Zygmund inequality for $\tilde{\eta}_{T+1,2T}(\varphi) - \mathbb{E}[\tilde{\eta}_{T+1,2T}(\varphi)|\mathcal{F}_{T+1}]$. The term $\mathbb{E}[\tilde{\eta}_{T+1,2T}(\varphi)|\mathcal{F}_{T+1}]$ will converge to $\tilde{\eta}_{T+1,2T}(\varphi)$ using the conditional i.i.d. property and the fact that $\tilde{\eta}_{T,T}^N(K_N(\theta)f_{\theta}(x_{T+1}^T))$ converges in probability to $\zeta(\theta, x_{T+1})$.

For $\tilde{\eta}_{T+1,2T}(\varphi)$, we consider:

$$
\tilde{\eta}_{T+1,2T}(\varphi) - \Phi_{T+2,2T,\tilde{\eta}_{T+1,2T}^N(\tilde{\eta}_{T+1,2T}^N)(\varphi)} + \Phi_{T+2,2T,\tilde{\eta}_{T+1,2T}^N(\tilde{\eta}_{T+1,2T}(\varphi))} - \tilde{\eta}_{T+1,2T}(\varphi)
$$

The first term is dealt with via the (conditional) Marcinkiewicz-Zygmund inequality as above. For $\Phi_{T+2,2T,\tilde{\eta}_{T+1,2T}^N(\tilde{\eta}_{T+1,2T}(\varphi))} - \tilde{\eta}_{T+1,2T}(\varphi)$ as $\tilde{\eta}_{T+1,2T}^N(\tilde{\eta}_{T+1,2T}(\varphi))$ converges in probability to $\tilde{\eta}_{T+1,2T}(\tilde{G}_{T+1})$, we need only consider

$$
\tilde{\eta}_{T+1,2T}^N(\tilde{G}_{T+1}\tilde{M}_{T+2,2T,\tilde{\eta}_{T+1,2T}(\varphi)}) - \tilde{\eta}_{T+1,2T}(\tilde{G}_{T+1}\tilde{M}_{T+2,2T,\tilde{\eta}_{T+1,2T}(\varphi)})
$$

The last term on the R.H.S. converges to zero by the above calculations. So we focus on the first term on the R.H.S. we have

$$
\mathbb{E}[\tilde{\eta}_{T+1,2T}^N(\tilde{G}_{T+1}\tilde{M}_{T+2,2T,\tilde{\eta}_{T+1,2T}(\varphi)}) - \tilde{\eta}_{T+1,2T}(\tilde{G}_{T+1}\tilde{M}_{T+2,2T,\tilde{\eta}_{T+1,2T}(\varphi)})]
$$

Let $\epsilon > 0$ be given. By (A4) there exists $\delta > 0$ independent of $\tilde{\xi}^T_{T+1}$ such that for any probability density $\eta$ with $|\eta - \pi_{T+1,2T}| < \delta$ we have that $|\tilde{M}_{T+2,2T,\tilde{\eta}_{T+1,2T}(\varphi)}(\tilde{\xi}^T_{T+1}) -
\( \hat{M}_{T+2.2T, \pi_{T+1,2T}}(\varphi)(\xi^1_{T+1}) \) < \epsilon/2. Consider the event:

\[ A(N, \delta) = \{ |\hat{\pi}_{T+1,2T} - \pi_{T+1,2T}| < \delta \}. \]

Then

\[
\mathbb{E}[|\hat{M}_{T+2.2T, \pi_{T+1,2T}}(\varphi)(\xi^1_{T+1}) - \hat{M}_{T+2.2T, \pi_{T+1,2T}}(\varphi)(\xi^1_{T+1})|] \\
= \mathbb{E}[|\hat{M}_{T+2.2T, \pi_{T+1,2T}}(\varphi)(\xi^1_{T+1}) - \hat{M}_{T+2.2T, \pi_{T+1,2T}}(\varphi)(\xi^1_{T+1})|]_{\mathbb{P}(A(N, \delta))} \\
+ \mathbb{E}[|\hat{M}_{T+2.2T, \pi_{T+1,2T}}(\varphi)(\xi^1_{T+1}) - \hat{M}_{T+2.2T, \pi_{T+1,2T}}(\varphi)(\xi^1_{T+1})|]_{\mathbb{P}(A(N, \delta)^c)} \\
\leq \epsilon/2 + C\mathbb{P}(A(N, \delta)^c).
\]

By the convergence in probability of \( \hat{\pi}_{T,T}^N(K_N(\theta) f_0(x_{T+1} | \cdot)) \) here is an \( N_0 \geq 1 \) such that for each \( N \geq N_0 \) we have \( 2C \cdot \mathbb{P}[A(N, \delta)^c] \leq \epsilon/2 \). Hence, for any \( N \geq N_0 \):

\[
\mathbb{E}[|\hat{G}_{T+1}(\xi^1_{T+1})| |\hat{M}_{T+2.2T, \pi_{T+1,2T}}(\varphi)(\xi^1_{T+1}) - \hat{M}_{T+2.2T, \pi_{T+1,2T}}(\varphi)(\xi^1_{T+1})|] \leq C\epsilon
\]

and as \( \epsilon > 0 \) was arbitrary, the term of interest goes to zero in \( L_1 \); this completes the proof. The proof can also be repeated if one considers a \( K_N(\theta - \theta') \) as part of the function (the argument is almost the same). The proofs at subsequent times follow the above arguments and are omitted for brevity.

**Appendix B: Proofs for Bias**

In the context of the proof for the bias, we need only consider one block (as will become apparent in the proof), as blocks are independent in the asymptotic bias. In addition, one significantly simplify the notations by simply considering two Feynman-Kac formula of \( T \) steps, with different initial distributions, the same potentials and different Markov kernels on measurable spaces \( (E_0, \mathcal{E}_0), \ldots, (E_T, \mathcal{E}_T) \). Thus, for \( k \in \{1, 2\} \) the two Feynman-Kac \( n \)-time marginals:

\[
\eta^k_n(dx_n) = \frac{\gamma^k_n(dx_n)}{\gamma^k_n(1)}
\]

with

\[
\gamma^k_n(dx_n) = \int_{\mathbb{R}^{d(n-1)}} \left\{ \prod_{p=0}^{n-1} G_p(x_p) M^k_{p+1}(x_p, dx_{p+1}) \right\} \eta^k_0(dx_0).
\]

This corresponds to our case, as the potentials are the same, with the Markov kernels and initial distributions different. Our proofs will depend a lot on the Bayes rule, which we now recall, for \( \mu \in \mathcal{P}(E_{p-1}) \)

\[
\Phi^k_p(\mu)(dx) = \frac{\mu(G_{p-1} M^k_{p}(dx))}{\mu(G_{p-1})}.
\]

We use the notation \( \Phi^k_{p,q}(\mu) := \Phi^k_q \circ \Phi^k_{q} \circ \cdots \circ \Phi^k_{p+1}(\mu) \), \( q \geq p \geq 0 \) (with the convention when \( p = q \), one returns \( \mu \)). Our assumption (A5) under the modified notation is

- There exist a \( \delta \in [1, \infty) \) such that for every \( p \geq 0 \)

\[
\sup_{x,y \in E_p} \frac{G_p(x)}{G_p(y)} \leq \delta.
\]
We have the standard telescoping sum, for \( \epsilon, \delta \) and for \( k \in \{1, 2\}, T - 1 \geq p \geq 1 \) every \( x, y \in E_{p-1} \)

\[
M^k_p(x, \cdot) \geq \epsilon \left( M^k_p(y, \cdot) \lor v(\cdot) \right).
\]

Recall that the final Markov kernel is a Dirac measure.

**Proof of Theorem 3.2** We have that, under our modified notations

\[
\mathcal{B}(T, \varphi) = |\Phi^1_T(\eta_0)(\varphi) - \Phi^2_T(\eta_0)(\varphi)| \leq |\Phi^1_T(\eta_0)(\varphi) - \Phi^1_T(\eta_0)(\varphi)| + |\Phi^1_T(\eta_0)(\varphi) - \Phi^2_T(\eta_0)(\varphi)|
\]

By Lemma B.2

\[
|\Phi^1_T(\eta_0)(\varphi) - \Phi^1_T(\eta_0)(\varphi)| \leq \|\varphi\|_\infty 4 \left( \frac{\delta}{\epsilon} \right)^2 (1 - \epsilon^2)^{T-1}.
\]

Then

\[
|\Phi^1_T(\eta_0^2)(\varphi) - \Phi^2_T(\eta_0^2)(\varphi)| = |\Phi^2_T(\Phi^1_T(\eta_0^2))(\varphi) - \Phi^2_T(\Phi^2_T(\eta_0^2))(\varphi)|,
\]

as \( \Phi^1_T(\mu)(\varphi) = \Phi^2_T(\mu)(\varphi) \) for any \( \mu \in \mathcal{P}(E_{T-1}) \). Now

\[
\Phi^2_T(\Phi^1_T(\eta_0^2))(\varphi) - \Phi^2_T(\Phi^2_T(\eta_0^2))(\varphi) = \frac{\Phi^2_T(\Phi^1_T(\eta_0^2))(\varphi) - \Phi^1_T(\eta_0^2)(\varphi)}{\Phi^1_T(\eta_0^2)(\varphi)} (G_{T-1} - \varphi)
\]

\[
+ \frac{\Phi^2_T(\Phi^2_T(\eta_0^2))(\varphi) - \Phi^2_T(\eta_0^2)(\varphi)}{\Phi^2_T(\eta_0^2)(\varphi)} (G_{T-1} - \varphi)
\]

\[
\times \left[ \Phi^2_T(\Phi^1_T(\eta_0^2))(\varphi) - \Phi^1_T(\eta_0^2)(\varphi) \right].
\]

Then, by application of Proposition B.1 along with (A5) it follows that

\[
|\Phi^2_T(\Phi^1_T(\eta_0^2))(\varphi) - \Phi^2_T(\Phi^2_T(\eta_0^2))(\varphi)| 
\]

\[
\leq 4\delta^2 \|\varphi\|_\infty \left( \frac{\delta^2}{\epsilon^4} \right)^2 (1 - (1 - \epsilon^2)^{T-1}) + 4\delta^4 \|\varphi\|_\infty \left( \frac{\delta^2}{\epsilon^4} \right)^2 (1 - (1 - \epsilon^2)^{T-1})
\]

which allows one to conclude the proof.

**Remark B.1** As one can see from inspection of the proof, the difference in initial distribution does not impact the bound. Moreover, the difference in Markov kernels is controlled, leading to a control of the bias; such a latter property is not obvious a priori and needs to be proved. As is evident from the proof, it does not matter which block one considers, under our assumptions.

**Proposition B.1** Assume (A5). Then for any \( \mu \in \mathcal{P}(E_0) \)

\[
\|\Phi^1_T(\mu) - \Phi^2_T(\mu)\|_\nu \leq 2 \left( \frac{\delta^2}{\epsilon^4} \right)^2 (1 - (1 - \epsilon^2)^{T-1})
\]

where \( \epsilon, \delta \) are as in (A5).

**Proof** We have the standard telescoping sum, for \( \varphi : E_{T-1} \to [0, 1] \)

\[
[\Phi^1_T(\mu) - \Phi^2_T(\mu)](\varphi)
\]

\[
= \sum_{k=0}^{T-1} \left[ \Phi^1_{k,T-1}(\Phi^2_\mu) \right] - \Phi^1_{k+1,T-1}(\Phi^2_{k+1}(\mu))](\varphi).
\]
Now, by Lemma B.2
\[ \left\| \Phi_{k,T-1}(\Phi_k^2(\mu)) - \Phi_{k+1,T-1}(\Phi_k^2(\mu)) \right\|_{tv} \leq 2\left( \frac{\delta}{\epsilon} \right)^2 (1 - \epsilon^2)^T \left\| \Phi_{k+1}^1(\Phi_k^2(\mu)) - \Phi_{k+1}^2(\Phi_k^2(\mu)) \right\|_{tv}. \]
By Lemma B.1
\[ \left\| \Phi_{k+1}^1(\Phi_k^2(\mu)) - \Phi_{k+1}^2(\Phi_k^2(\mu)) \right\|_{tv} \leq (1 - \epsilon) \]
so we have proved that
\[ \left\| \Phi_{pl,(p+1)}^1(\mu) - \Phi_{pl,(p+1)}^2(\mu) \right\|_{tv} \leq 2\left( \frac{\delta}{\epsilon} \right)^2 (1 - \epsilon)^2 \sum_{k=0}^{T-1} (1 - \epsilon^2)^{T-k-2}, \]
from which one can easily conclude.

\textbf{Lemma B.1} Assume (A5). Then for any \( T - 1 \geq p \geq 1, \mu \in \mathcal{P}(E_{p-1}) \) and \( \varphi : \mathbb{R}^d \to [0, 1] \) we have
\[ |[\Phi_p^1(\mu) - \Phi_p^2(\mu)](\varphi)| \leq (1 - \epsilon) \]
where \( \epsilon \) is as in (A5).

\textbf{Proof} We have
\[ [\Phi_p^1(\mu) - \Phi_p^2(\mu)](\varphi) = \frac{1}{\mu(G_{p-1})} \mu(G_{p-1}[M_p^1 - M_p^2](\varphi)). \]
Now, by (A5), for any \( x, k \) one can write
\[ M_p^k(x, dy) = (1 - \epsilon)R_p^k(x, dy) + \epsilon \nu(dy) \]
where \( R_p^k(x, dy) \) is a Markov kernel:
\[ R_p^k(x, dy) = \frac{M_p^k(x, dy) - \epsilon \nu(dy)}{(1 - \epsilon)}. \]
Thus
\[ [M_p^1 - M_p^2](\varphi)(x) = (1 - \epsilon)[R_p^1 - R_p^2](\varphi)(x) \]
from which one easily concludes.

The following Lemma just collects some results of Del Moral (2004) into a convenient form for use in the above proofs. We make the following definitions for \( 0 \leq s < t, k \in \{1, 2\} \): \( Q_{s,t}^k(x, dy) = G_{s-1}(x)M_s^k(x, dy), \)
\[ Q_{s,t}^k(x_p, dx_n) = \int_{\mathbb{R}^d(s-2)} \prod_{q=s+1}^t Q_q^k(x_{q-1}, dx_q) \]
and \( P_{s,t}^k(x, dy) = Q_{s,t}^k(x, dy)/Q_{s,t}^k(1)(x) \). For a bounded and measurable real-valued function \( \varphi \) we denote
\[ \|\varphi\|_{osc} = \sup_x |\varphi(x)| + \sup_{x,y} |\varphi(x) - \varphi(y)|. \]
Lemma B.2 Assume (A5). Then for any $0 \leq s < t \leq T - 1$ and any $\mu, \rho \in \mathcal{P}(E_{s-1})$, $k \in \{1, 2\}$

$$\|\Phi^k_{s,t}(\mu) - \Phi^k_{s,t}(\rho)\|_{tv} \leq 2\left(\frac{\delta}{\epsilon}\right)^2 (1 - \epsilon^2)^{t-s} \|\mu - \rho\|_{tv}$$

where $\epsilon, \delta$ are as in (A5).

Proof By (Del Moral 2004, Theorem 4.3.1)

$$\|\Phi^k_{s,t}(\mu) - \Phi^k_{s,t}(\rho)\|_{tv} \leq \beta(P^k_{s,t}) \frac{\|Q^k_{s,t}(1)\|_{osc}}{\mu(Q^k_{s,t}(1)) \wedge \rho(Q^k_{s,t}(1))} \|\mu - \rho\|_{tv}$$

where $\beta(P^k_{s,t})$ is the Dobrushin coefficient of $P^k_{s,t}$. Then by (Del Moral 2004, pp. 142) (fourth displayed equation) and (A5)

$$\beta(P^k_{s,t}) \leq (1 - \epsilon^2)^{t-s}.$$ 

Note that via (A5) and (Cérou et al. 2011, Lemma 4.1) one has

$$\sup_{x,y} \frac{Q^k_{s,t}(1)(x)}{Q^k_{s,t}(1)(y)} \leq \frac{\delta}{\epsilon}$$

and as a result via (Del Moral 2004, pp. 138), for any pair of probabilities $\mu, \rho \in \mathcal{P}(E_{s-1})$

$$\frac{\|Q^k_{s,t}(1)\|_{osc}}{\mu(Q^k_{s,t}(1)) \wedge \rho(Q^k_{s,t}(1))} \leq 2\left(\frac{\delta}{\epsilon}\right)^2.$$ 

The result thus follows. \qed

References

Andrieu C, Doucet A, Holenstein R (2010) Particle Markov chain Monte Carlo methods (with discussion). J R Statist Soc Ser B 72:269–342
Andrieu C, Doucet A, Tadić V (2009) On-line parameter estimation in general state-space models using pseudo-likelihood. Unpublished Technical Report
Beskos A, Jasra A, Kantas N, Thiery A (2016) On the convergence of adaptive sequential Monte Carlo methods. Ann Appl Probab 26:1111–1146
Borwanker J, Kallianpur G, Prakasa Rao BLS (1971) The Bernstein-Von Mises theorem for Markov processes. Ann Math Stat 42:1241–1253
Cappé O, Ryden T, Moulines É (2005) Inference in hidden Markov models. Springer, New York
Centanni S, Minozzo M (2006) A Monte Carlo approach to filtering for a class of marked doubly stochastic Poisson processes. J Amer Stat Assoc 101:1582–1597
Cérou F, Del Moral P, Guyader A (2011) A non-asymptotic variance theorem for un-normalized Feynman-Kac particle models. Ann Inst Henri Poincare 47:629–649
Chan HP, Heng CW, Jasra A (2016) Theory of parallel particle filters for hidden Markov models. Adv Appl Probab 48:69–87
Chopin N, Jacob P, Papaspiliopoulos O (2013) SMC$^2$: a sequential Monte Carlo algorithm with particle Markov chain Monte Carlo updates. J R Stat Soc B 75:397–426
Crisan D, Miguez J (2014) Particle-kernel estimation of the filter density in state-space models. Bernoulli 20:1879–1929
Crisan D, Miguez J (2014) Nested particle filters for online parameter estimation in discrete-time state-space Markov models. arXiv preprint
Deligiannidis G, Doucet A, Pitt MK (2015) The correlated pseudo-marginal method. arXiv preprint
Del Moral P (2004) Feynman-Kac formulae: genealogical and interacting particle systems with applications. Springer, New York
Del Moral P (2013) Mean field simulation for Monte Carlo integration. Chapman & Hall, London
Del Moral P, Doucet A, Jasra A (2006) Sequential Monte Carlo samplers. J R Stat Soc B 68:411–436
Douc R, Moulines E, Olsson J, Van Handel R (2011) Consistency of the maximum likelihood estimator for general hidden Markov models. Ann Stat 39:474–513
Doucet A, Johansen A (2011) A tutorial on particle filtering and smoothing: fifteen years later. In: Crisan et B, Rozovskiy D (eds) Handbook of nonlinear filtering. Oxford University Press, Oxford
Fearnhead P (2002) MCMC, sufficient statistics and particle filters. J Comp Graph Stat 11:848–862
Gilks WR, Berzuini C (2001) Following a moving target - Monte Carlo inference for dynamic Bayesian models. J R Stat Soc B 63:127–146
Jacob P, Murray L, Rubenthaler S (2015) Path storage in the particle filter. Stat Comp 25:487–496
Kantas N, Doucet A, Singh SS, Maciejowski JM, Chopin N (2015) On particle methods for parameter estimation in state-space models. Stat Sci 30:328–351
Polson NG, Stroud JR, Müller P (2008) Practical filtering with sequential parameter learning. J R Stat Soc B 70:413–428