A simple scheme for the parallelization of particle filters and its application to the tracking of complex stochastic systems

Katrin Achutegui*  Dan Crisan†  Joaquín Míguez‡  Gonzalo Ríos§

July 31, 2014

Abstract

Considerable effort has been devoted in the past decade to the design of schemes for the parallel, or distributed, implementation of particle filters. The approaches vary from the totally heuristic to the mathematically well-principled. However, the former are largely based on (often loose) approximations that prevent the claim of any rigorous guarantees of convergence, whereas the latter involve considerable overhead to ensure the proper interaction of particles, which impairs the efficiency of the intended parallelization. In this paper we investigate the use of possibly the simplest scheme for the parallelization of the standard particle filter, that consists in splitting the computational budget into $M$ fully independent particle filters with $N$ particles each, and then obtaining the desired estimators by averaging over the $M$ independent outcomes of the filters. This approach minimizes the parallelization overhead yet displays highly desirable theoretical properties. In particular, we prove that the mean square estimation error (of integrals with respect to the filter measure) vanishes asymptotically with the same rate, proportional to $1/MN$, as the standard (centralized) particle filter with the same total number of particles. Parallelization, therefore, has the obvious advantage of dividing the running times while preserving the (asymptotic) performance. We also propose a time-error index to quantify this improvement and to compare schemes with different degrees of parallelization. As a side result, we show that the expected value of the random probability measure output by each independent filter converges in total variation distance to the true posterior with rate of order $1/N$ (note that the average measure over the $M$ filters is just a sample-mean estimate of this expected measure). Finally, we provide an illustrative numerical example for the tracking of a Lorenz 65 chaotic system with dynamical noise and partial (noisy) observations. Our computer simulations show how, in a practical application, the proposed parallelization scheme can attain the same approximation accuracy as the corresponding centralized particle filter with only a small fraction of the running time.

*Department of Signal Theory & Communications, Universidad Carlos III (Spain). E-mail: kachutegui@tsc.uc3m.es.
†Department of Mathematics, Imperial College London (UK). E-mail: d.crisan@imperial.ac.uk.
‡Department of Signal Theory & Communications, Universidad Carlos III (Spain). E-mail: joaquin.miguez@uc3m.es.
§Department of Signal Theory & Communications, Universidad Carlos III (Spain). E-mail: grios@tsc.uc3m.es.
1 Introduction

Over the past decade there has been a continued interest in the design of schemes for the implementation of particle filtering algorithms using parallel or distributed hardware of various types, including general purpose devices such as multi-core CPUs or graphical processing units (GPUs), and application-tailored devices such as field programmable gate arrays (FPGAs).

A particle filter is a recursive algorithm for the approximation of a sequence of posterior probability measures arising from stochastic dynamical systems in state-space form (see, e.g., [10, 3, 11, 16] and references therein for a general view of the field). A typical particle filter includes three steps that are repeated sequentially:

- Monte Carlo sampling in the space of the state variables,
- computation of weights for the generated samples and, finally,
- resampling according to the weights.

While at first sight the algorithm may look like a straightforward to parallelize (sampling and weighting can be carried out concurrently without any constraint), the resampling step involves the interaction of the whole set of Monte Carlo samples. Several authors have proposed schemes for ‘splitting’ the resampling step into simpler tasks that can be carried out concurrently. The approaches are diverse and range from the heuristic [12, 18, 15] to the mathematically well-principled [23, 22] (see also [2] together with [19]). However, the former are largely based on (often loose) approximations that prevent the claim of any rigorous guarantees of convergence, whereas the latter involve non-negligible overhead to ensure the proper interaction of particles. The ‘island’ particle method of [22], for example, performs resampling at two levels (involving individual particles and sets of particles, respectively) and only one of the two admits direct parallelization. The distributed resampling scheme of [2, 19] is similar, as it resamples small subsets of particles in parallel, but the second level of resampling is substituted by an exchange of particles among different subsets which are typically assigned to different processing elements. The idea of exchanging particles is generalized in the α sequential Monte Carlo (α-SMC) methodology [23], where the resampling step is parameterized by designing a sequence of maps of interactions among particles. The higher-level resampling step of [22], the particle exchange of [2] or the parameterized interaction of [23] imply a computational overhead, i.e., there are extra computations that have to be performed in exchange for parallelizing the original resampling task.

In this paper we investigate the use of possibly the simplest scheme for the parallelization of the standard particle filter, that consists in splitting the computational budget into $M$ fully independent particle filters with $N$ particles each, and then obtaining the desired estimators by averaging over the $M$ independent outcomes of the filters. This approach minimizes the parallelization overhead, since there is no interaction at all among the filters, yet displays desirable theoretical properties. In particular, we use
a variance–bias argument to prove that the mean square approximation error for integrals of bounded functions with respect to the filter measure vanish asymptotically with the same rate (proportional to $1/MN$) as in the standard (centralized) particle filter with the same total number of particles $K = MN$. Independent parallelization, therefore, has the obvious advantage of reducing the running time by a factor of $M$ (depending on the kind of hardware being used) while preserving the (asymptotic) performance. To illuminate this idea and quantify the improvement, we introduce a time-error index that brings together the time complexity (asymptotic order of the running time) and the estimation accuracy (asymptotic error rates) into a single quantitative figure of merit that can be used to compare schemes with different degrees of parallelization. As a side result, we also show that the expected value of the random probability measure output by each independent filter converges in total variation distance to the true posterior with rate of order $1/N$ (and we note that the average measure over the $M$ filters is just a sample-mean estimate of this expected measure). We also present an illustrative example for the tracking of a stochastic Lorenz 65 chaotic system with partial (and noisy) observations, using a pool of multi-core CPUs for the implementation of the parallel filters. The numerical results, involving both estimation errors and running times, provide numerical evidence that support the theoretical findings.

The analysis of the particle filters in this paper is based on very mild assumptions on the stochastic dynamic model and classic induction arguments. We do not address uniform convergence over time or impose further assumptions on the models because our aim is to illuminate the relationship between the standard (centralized) particle filter and its parallelized versions in the simplest possible framework. A more sophisticated analysis can obviously be carried out by imposing additional assumptions on the dynamic models, yet the underlying argument for the comparison of parallel schemes would be identical. Furthermore, while we have focused here on particle filters for discrete-time state-space models, the analysis can be similarly done for continuous-time systems and, indeed, the basic results needed for that case (the analysis of the approximation) can be found in [14].

The rest of the paper is organized as follows. In Section 2, we present basic background material and notations to be used through the rest of the paper. The analytical results are introduced in Section 3, including the analysis of the bias, the mean square error and the total variation distance, as well as the proposed time-error index. Numerical results for the tracking of a stochastic Lorenz 65 system are displayed in Section 4, and finally, Section 5 is devoted to a discussion and some concluding remarks.

2 Background

2.1 Notation and preliminaries

We first introduce some common notations to be used through the paper, broadly classified by topics. Below, $\mathbb{R}$ denotes the real line, while for an integer $d \geq 1$, $\mathbb{R}^d = \mathbb{R} \times \ldots \times \mathbb{R}$.
• Functions.
  - The supremum norm of a real function $f : \mathbb{R}^d \to \mathbb{R}$ is denoted as $\|f\|_\infty = \sup_{x \in \mathbb{R}^d} |f(x)|$.
  - $B(\mathbb{R}^d)$ is the set of bounded real functions over $\mathbb{R}^d$, i.e., $f \in B(\mathbb{R}^d)$ if, and only if, $\|f\|_\infty < \infty$.

• Measures and integrals. Let $S \subseteq \mathbb{R}^d$ be a subset of $\mathbb{R}^d$.
  - $B(S)$ is the $\sigma$-algebra of Borel subsets of $S$.
  - $\mathcal{P}(S)$ is the set of probability measures over the measurable space $(B(S), S)$.
  - $(f, \mu) \triangleq \int f(x)\mu(dx)$ is the integral of a real function $f : S \to \mathbb{R}$ w.r.t. a measure $\mu \in \mathcal{P}(S)$.
  - Given a probability measure $\mu \in \mathcal{P}(S)$, a Borel set $A \in B(S)$ and the indicator function

\[
I_A(x) = \begin{cases} 
1, & \text{if } x \in A \\
0, & \text{otherwise}
\end{cases} ,
\]

$\mu(A) = (I_A, \mu) = \int I_A(x)\mu(dx)$ is the probability of $A$.

• Sequences, vectors and random variables (r.v.).
  - We use a subscript notation for sequences, namely $x_{t_1:t_2} \triangleq \{x_{t_1}, \ldots, x_{t_2}\}$.
  - For an element $x = (x_1, \ldots, x_d) \in \mathbb{R}^d$ of an Euclidean space, its norm is denoted as $\|x\| = \sqrt{x_1^2 + \ldots + x_d^2}$.
  - The $L_p$ norm of a real r.v. $Z$, with $p \geq 1$, is written as $\|Z\|_p \triangleq E[|Z|^p]^{1/p}$, where $E[\cdot]$ denotes expectation w.r.t. the distribution of $Z$.

2.2 State-space Markov models in discrete time

Consider two random sequences, $\{X_t\}_{t \geq 0}$ and $\{Y_t\}_{t \geq 1}$, taking values in $\mathcal{X} \subseteq \mathbb{R}^{d_x}$ and $\mathbb{R}^{d_y}$, respectively. Let $P_1$ be the joint probability measure for the collection of random variables $(X_0, X_s, Y_n)_{1 \leq s \leq t}$.

We refer to the sequence $\{X_t\}_{t \geq 0}$ as the state (or signal) process and we assume that it is an inhomogeneous Markov chain governed by an initial probability measure $\tau_0 \in \mathcal{P}(\mathcal{X})$ and a sequence of Markov kernels $\tau_t : \mathcal{B}(\mathcal{X}) \times \mathcal{X} \to [0, 1]$. To be specific, we define

\[
\tau_0(A) \triangleq P_0 \{X_0 \in A\} ,
\]

\[
\tau_t(A|x_{t-1}) \triangleq P_t \{X_t \in A|X_{t-1} = x_{t-1}\} , \quad t \geq 1 ,
\]

where $A \in \mathcal{B}(\mathcal{X})$ is a Borel set. The sequence $\{Y_t\}_{t \geq 1}$ is termed the observation process. Each r.v. $Y_t$ is assumed to be conditionally independent of other observations given $X_t$, namely

\[
P_t \{Y_t \in A|X_{0:t} = x_{0:t}, \{Y_k = y_k\}_{k \neq t}\} = P_t \{Y_t \in A|X_t = x_t\}
\]

for any $A \in \mathcal{B}(\mathbb{R}^{d_y})$. Additionally, we assume that every probability measure $\gamma_t \in \mathcal{P}(\mathbb{R}^{d_y})$ in the family

\[
\gamma_t(A|x_t) \triangleq P_t \{Y_t \in A|X_t = x_t\} , \quad A \in \mathcal{B}(\mathbb{R}^{d_y}) , \quad t \geq 1 ,
\]

\[4]
has a nonnegative density w.r.t. the Lebesgue measure, denoted $g_t(y|x_t) > 0$, hence we can write
\[
\gamma_t(A|x_t) = \int I_A(y)g_t(y|x_t)dy.
\]
(4)

We often use $g_t$ as a function of $x_t$ (hence, as a likelihood) and we emphasize this by writing $g_t^g(x) \triangleq g_t(y|x)$. The prior $\tau_0$, the kernels $\{\tau_t\}_{t \geq 1}$, and the functions $\{g_t\}_{t \geq 1}$, describe a stochastic Markov state-space model in discrete time.

The stochastic filtering problem consists in the computation of the posterior probability measure of the state $X_t$ given the sequence of observations up to time $t$. Specifically, for a given observation record $\{y_t\}_{t \geq 1}$, we seek the measures
\[
\pi_t(A) \triangleq \mathbb{P}_t\{X_t \in A|Y_{1:t} = y_{1:t}\}, \quad t = 0, 1, 2, ...
\]
where $A \in \mathcal{B}(\mathcal{X})$. For many practical problems, the interest actually lies in the computation of integrals of the form $(f, \pi_t)$ for some integrable function $f : \mathcal{X} \to \mathbb{R}$. Note that, for $t = 0$, we recover the prior signal measure, i.e., $\pi_0 = \tau_0$.

An associated problem is the computation of the one-step-ahead predictive measure
\[
\xi_t(A) \triangleq \mathbb{P}_t\{X_t \in A|Y_{1:t-1} = y_{1:t-1}\}, \quad t = 1, 2, ...
\]
This measure can be explicitly written in terms of the kernel $\tau_t$ and the filter $\pi_{t-1}$. Indeed, for any integrable function $f : \mathcal{X} \to \mathbb{R}$, we readily obtain (see, e.g., [1, Chapter 10])
\[
(f, \xi_t) = \int \int f(x)\tau_t(dx|x')\pi_{t-1}(dx') = ((f, \tau_t), \pi_{t-1}),
\]
and we write $\xi_t = \tau_t\pi_t$ as shorthand.

The filter at time $t$, $\pi_t$, can be obtained from the predictive measure, $\xi_t$, and the likelihood, $g_t^y$, by way of the so-called projective product [1] $\pi_t = g_t^y \ast \xi_t$, defined as
\[
(f, g_t^y \ast \xi_t) \triangleq \frac{(f \cdot g_t^y, \xi_t)}{(g_t^y, \xi_t)}
\]
for any integrable function $f : \mathbb{R}^d x \to \mathbb{R}$, which, combined with (5), yields the recursive formula
\[
\pi_t = g_t^y \ast \tau_t\pi_{t-1}.
\]
(6)

It is also possible to think of the sequence of measures $\{\rho_t\}_{t \geq 0}$, where
\[
\rho_0 = \pi_0, \quad \rho_t = g_t^y \ast \tau_t\rho_{t-1}
\]
and, for any integrable $f : \mathcal{X} \to \mathbb{R}$ and $\alpha \in \mathcal{P}(\mathcal{X})$, we define
\[
(f, g_t^y \cdot \alpha) \triangleq (f \cdot g_t^y, \alpha).
\]
(8)

Note that $\rho_t$ is not a probability measure, but an unnormalized version of $\pi_t$, namely
\[
(f, \pi_t) = \frac{(f, \rho_t)}{(1, \rho)}
\]
where $1(x) = 1$ is the constant unit function.
2.3 Standard particle filter

Assume that a sequence of observations \( Y_{1:T} = y_{1:T} \), for some \( T < \infty \), is given. Then, the sequences of measures \( \{\pi_t\}_{t \geq 1} \), \( \{\xi_t\}_{t \geq 1} \) and \( \{\rho_t\}_{t \geq 0} \) can be numerically approximated using particle filtering. Particle filters are numerical methods based on the recursive relationships (6) and (8). The simplest algorithm, often called ‘standard particle filter’ or ‘bootstrap filter’ \([13]\) (see also \([9]\)), can be described as follows.

**Algorithm 1** Bootstrap filter.

1. **Initialization.** At time \( t = 0 \), draw \( N \) i.i.d. samples, \( x_0^{(i)} \), \( i = 1, \ldots, N \), from the distribution \( \tau_0 \).

2. **Recursive step.** Let \( \{x_{t-1}^{(i)}\}_{1 \leq i \leq N} \) be the particles (samples) generated at time \( t-1 \). At time \( t \), proceed with the two steps below.

   (a) For \( i = 1, \ldots, N \), draw a sample \( \bar{x}_t^{(i)} \) from the probability distribution \( \tau_t(\cdot|x_{t-1}^{(i)}) \) and compute the normalized weight
   \[
   w_t^{(i)} = \frac{g_t^{(i)}(\bar{x}_t^{(i)})}{\sum_{k=1}^N g_t^{(k)}(x_t^{(k)})}, \tag{9}
   \]

   (b) For \( i = 1, \ldots, N \), let \( x_t^{(i)} = \bar{x}_t^{(k)} \) with probability \( w_t^{(k)} \), \( k \in \{1, \ldots, N\} \).

Step 2.(b) is referred to as resampling or selection. In the form stated here, it reduces to the so-called multinomial resampling algorithm \([11, 8]\) but the convergence of the filter can be easily proved for various other schemes (see, e.g., the treatment of the resampling step in \([5]\)).

Using the sets \( \{\bar{x}_t^{(i)}\}_{1 \leq i \leq N} \) and \( \{x_t^{(i)}\}_{1 \leq i \leq N} \), we construct random approximations of \( \xi_t \), \( \rho_t \) and \( \pi_t \), namely
\[
\xi_N^t = \frac{1}{N} \sum_{i=1}^N \delta_{\bar{x}_t^{(i)}}, \quad \pi_N^t = \frac{1}{N} \sum_{i=1}^N \delta_{x_t^{(i)}}, \quad \text{and} \quad \rho_N^t = G_N^t \pi_N^t \tag{10}
\]
where \( \delta_x \) is the delta unit-measure located at \( x \in \mathbb{R}^d \) and
\[
G_N^t = \frac{1}{N^d} \prod_{k=1}^t \left( \sum_{j=1}^N g_k^{(j)}(\bar{x}_k^{(j)}) \right). \tag{11}
\]

For any integrable function \( f \) on the state space, it is straightforward to approximate the integrals \( (f, \xi_t) \), \( (f, \pi_t) \) and \( (f, \rho_t) \) as
\[
(f, \xi_t) \approx (f, \xi_N^t) = \frac{1}{N} \sum_{i=1}^N f(\bar{x}_t^{(i)}),
\]
\[
(f, \pi_t) \approx (f, \pi_N^t) = \frac{1}{N} \sum_{i=1}^N f(x_t^{(i)}), \quad \text{and}
\]
\[
(f, \rho_t) \approx (f, \rho_N^t) = G_N^t (f, \pi_N^t),
\]
respectively.
The convergence of particle filters has been analyzed in a number of different ways. Here we use simple results for the convergence of the $L_p$ norms ($p \geq 1$) of the approximation errors. For the approximation of integrals w.r.t. $\xi_t$ and $\pi_t$ we have the following standard result.

**Lemma 1** Assume that the sequence of observations $Y_{1:T} = y_{1:T}$ is fixed (with $T < \infty$), $g_t^{y_t} \in B(X)$ and $g_t^{y_t} > 0$ (in particular, $(g_t^{y_t}, \xi_t) > 0$) for every $t = 1, 2, ..., T$. Then for any $f \in B(X)$, any $p \geq 1$ and every $t = 1, \ldots, T$,

$$
\| (f, \xi_t^N) - (f, \xi_t) \|_p \leq \bar{c}_t \| f \|_\infty \sqrt{N} \quad \text{and} \quad \| (f, \pi_t^N) - (f, \pi_t) \|_p \leq c_t \| f \|_\infty \sqrt{N},
$$

(12)

where $\bar{c}_t$ and $c_t$ are finite constants independent of $N$, $\| f \|_\infty = \sup_{x \in X} |f(x)| < \infty$ and the expectations are taken over the distributions of the measure-valued random variables $\xi_t^N$ and $\pi_t^N$, respectively.

**Proof:** This result is a special case of, e.g., Lemma 1 in [20]. $\square$

**Remark 1** The constants $\bar{c}_t$ and $c_t$ can be easily shown to increase with $t$. It is possible to find error rates independent of $t$ by imposing additional assumptions on the state-space model (related to the stability of the optimal filter) [7].

3 Ensembles of independent particle filters

3.1 Overview

Assume we run $M$ independent bootstrap filters, with $N$ particles each, for the same sequence of observations $\{y_t\}_{0 \leq t \leq T}$. Each filter yields a random approximation $\pi_{t,M,N}^i$, $i = 1, \ldots, M$, from which we compute the average $\pi_{t,M,N}^i = \frac{1}{M} \sum_{i=1}^{M} \pi_{t,M,N}^i$ and adopt the mean square error (MSE) for bounded real test functions, $E \left[ \left( (f, \pi_{t,M,N}) - (f, \pi_t) \right)^2 \right]$, $f \in B(X)$, as a performance metric. Since the underlying state-space model is the same for all filters and they are run in a completely independent manner, the measured-valued random variables $\pi_{t,M,N}^i$, $i = 1, \ldots, M$, are i.i.d. and it is straightforward to show (via Lemma 1) that

$$
E \left[ \left( (f, \pi_{t,M,N}) - (f, \pi_t) \right)^2 \right] \leq \frac{\bar{c}_t^2 \| f \|_\infty^2}{MN},
$$

(13)

for some constant $t$ independent of $N$ and $M$. However, the inequality (13) falls short of characterizing the performance of the parallelization scheme because it does not display the effect of the choice of $N$. In the extreme case of $N = 1$, for example, $\pi_{t,M,N}^i$ reduces to the outcome of a sequential importance sampling algorithm, with no resampling, which is known to degenerate quickly in practice. Instead of (13), we seek a bound for the approximation error that provides some indication on the trade-off between the number of independent filters, $M$, and the number of particles per filter, $N$.

In this section we derive bounds for the approximation error $\left( (f, \pi_{t,M,N}) - (f, \pi_t) \right)^2$ based on the classical decomposition of the MSE in variance and bias terms. First, we obtain preliminary results
that are needed for the analysis of the average measure $\pi_{t}^{M \times N}$. In particular, we prove that the random unnormalized measure $\rho_{t}^{N}$ produced by the bootstrap filter (Algorithm 1) is unbiased and attains $L_{p}$ error rates proportional to $\frac{1}{\sqrt{N}}$, i.e., the same as $\xi_{t}^{N}$ and $\pi_{t}^{N}$. We use these results to derive an upper bound for the bias of $\pi_{t}^{N}$ which is proportional to $\frac{1}{\sqrt{N}}$. The latter, in turn, enables us to deduce an upper bound for the MSE of the ensemble approximation $\pi_{t}^{M \times N}$ consisting of two additive terms that depend explicitly on $M$ and $N$. If the total number of particles $K = NM$ is fixed, our results show that

- the (asymptotic) MSE rate and
- the running time of the filtering algorithm

are jointly minimize if we run $M = \sqrt{K}$ independent particle filters in parallel, with $N = \sqrt{K}$ particles each. In other words, if the performance metric is running time and the asymptotic MSE rate, the interaction among the $M$ filters (as advocated in the literature [2, 18, 23, 22]) does not bring any gain and should be avoided (to prevent the associated overhead).

All the results to be introduced in the rest of Section 3 hold true under the (mild) assumptions of Lemma 1, which we summarize below for convenience of presentation.

**Assumption 1** The sequence of observations $Y_{1:T} = y_{1:T}$ is arbitrary but fixed, with $T < \infty$.

**Assumption 2** The likelihood functions are bounded and positive, i.e.,

$$g_{t}^{y_{t}} \in B(\mathcal{X}) \quad \text{and} \quad g_{t}^{y_{t}} > 0 \quad \text{for every} \quad t = 1, 2, ..., T.$$ 

**Remark 2** Note that Assumptions 1 and 2 imply that

- $(g_{t}^{y_{t}}, \alpha) > 0$, for any $\alpha \in \mathcal{P}(\mathcal{X})$, and
- $\prod_{k=1}^{T} g_{k}^{y_{k}} \leq \prod_{k=1}^{T} \|g_{k}^{y_{k}}\|_{\infty} < \infty,$

for every $t = 1, 2, ..., T$.

**Remark 3** We seek simple convergence results for a fixed time horizon $T < \infty$, similar to Lemma 1. Therefore, no further assumptions related to the stability of the optimal filter for the state-space model [7] are needed. If such assumptions are imposed then stronger (time uniform) asymptotic convergence rates can be found, both for the independent approximate filters $\pi_{t}^{i,N}$ and their average $\pi_{t}^{M \times N}$.

### 3.2 Bias and error rates

We first show that the approximation of the unnormalized filter measure $\rho_{t}$ in Eq. (10) is unbiased.

**Lemma 2** If Assumptions 1 and 2 hold, then

$$E \left[ (f, \rho_{t}^{N}) \right] = (f, \rho_{t})$$

for any $f \in B(\mathcal{X})$ and every $t = 1, 2, ..., T$. 

Proof: We proceed by induction in the time index \( t \). For \( t = 0 \), \( \rho_0 = \pi_0 = \pi_0 \) and, since \( x_0^{(i)} \), \( i = 1, \ldots, N \), are drawn from \( \pi_0 \), the equality \( E[(f, \rho_{N}^{t})] = (f, \rho_{0}) \) is straightforward.

Let us assume that
\[
E \left[ (f, \rho_{t-1}^{N}) \right] = (f, \rho_{t-1})
\]
for some \( t > 0 \) and any \( f \in B(\mathcal{X}) \). If we use \( \bar{\mathcal{F}}_t \) to denote the \( \sigma \)-algebra generated by the set of random variables \( \{x_{0,t-1}^{(i)}, x_{1,t}^{(i)} : 1 \leq i \leq N\} \) then we readily find that
\[
E \left[ (f, \rho_{t}^{N}) \bigg| \bar{\mathcal{F}}_t \right] = E \left[ G^N_t(f, \pi_{t}^{N}) \bigg| \bar{\mathcal{F}}_t \right] = G^N_t(f, \bar{\pi}_{t}^{N}),
\]
since \( G^N_t \) is measurable w.r.t. \( \bar{\mathcal{F}}_t \) and \( E[(f, \pi_{t}^{N})] = (f, \bar{\pi}_{t}^{N}) \). Moreover, if we recall that
\[
(f, \bar{\pi}_{t}^{N}) = \sum_{i=1}^{N} \bar{w}_t^{(i)} f(x_{t}^{(i)}) = \sum_{i=1}^{N} \bar{g}_t^{(i)}(x_{t}^{(i)}) f(x_{t}^{(i)}) = \frac{(f, \bar{\pi}_{t}^{N})}{(g_t^{(i)}, \xi_t^{(i)})}
\]
then it is apparent from the definition of \( G^N_t \) in (14) that
\[
G^N_t(f, \bar{\pi}_{t}^{N}) = G^N_{t-1}(f, \bar{g}_t^{(i)}, \xi_t^{(i)}).
\]
Taking together (15) and (16) we have
\[
E \left[ (f, \rho_{t}^{N}) \bigg| \bar{\mathcal{F}}_t \right] = G^N_{t-1}(f, \bar{g}_t^{(i)}, \xi_t^{(i)}).
\]

Let \( \mathcal{F}_{t-1} \) be the \( \sigma \)-algebra generated by the set of variables \( \{x_{0,t-1}^{(i)}, x_{0,0}^{(i)} : 1 \leq i \leq N\} \). Since \( \mathcal{F}_{t-1} \subseteq \mathcal{F}_t \), Eq. (17) yields
\[
E \left[ (f, \rho_{t}^{N}) \bigg| \mathcal{F}_{t-1} \right] = E \left[ G^N_{t-1}(f, \bar{g}_t^{(i)}, \xi_t^{(i)}) \bigg| \mathcal{F}_{t-1} \right] = G^N_{t-1} E \left[ (f, \bar{g}_t^{(i)}, \xi_t^{(i)}) \bigg| \mathcal{F}_{t-1} \right],
\]

since \( G^N_{t-1} \) is measurable w.r.t. \( \mathcal{F}_{t-1} \). Moreover, for any \( h \in B(\mathcal{X}) \), it is straightforward to show that
\[
E[(h, \bar{\pi}_{t}^{N})] = (h, \tau_t \pi_{t-1}^{N}) = ((h, \tau_t), \pi_{t-1}^{N}),
\]
hence, as \( f g_t^{(i)} \in B(\mathcal{X}) \), we readily obtain
\[
E \left[ (f, \bar{\pi}_{t}^{N}) \bigg| \mathcal{F}_{t-1} \right] = ((f, \bar{g}_t^{(i)}, \tau_t), \pi_{t-1}^{N}).
\]
Substituting (15) into (18) we arrive at
\[
E \left[ (f, \rho_{t}^{N}) \bigg| \mathcal{F}_{t-1} \right] = G^N_{t-1} \left( (f, \bar{g}_t^{(i)}), \pi_{t-1}^{N} \right)
\]
\[
= \left( (f, \bar{g}_t^{(i)}), \rho_{t-1}^{N} \right),
\]
where (20) follows from the definition of the estimate of \( \rho_{t-1} \), namely \( \rho_{t-1}^{N} = G^N_{t-1} \pi_{t-1}^{N} \). If we take unconditional expectations on both sides of Eq. (20), we obtain
\[
E \left[ (f, \rho_{t}^{N}) \right] = E \left[ (f, \bar{g}_t^{(i)}), \rho_{t-1}^{N} \right]
\]
\[
= ((f, \bar{g}_t^{(i)}), \rho_{t-1})
\]
\[
= (f, \bar{g}_t^{(i)} \cdot \tau_t \rho_{t-1})
\]
\[
= (f, \rho_{t}),
\]
9
where equality \((21)\) follows from the induction hypothesis \((14)\), \((22)\) is obtained by simply re-ordering \((21)\) and Eq. \((23)\) follows from the recursive definition of \(\rho_t\) in \((7)\).

\[\Box\]

Combining Lemma \((2)\) with the standard result of Lemma \((1)\) enables us to obtain an explicit convergence rate for the \(L_p\) norms of the approximation errors \((f, \rho_t^N) - (f, \rho_t)\).

**Lemma 3** If Assumptions \((7)\) and \((3)\) hold, then, for any \(f \in B(\mathcal{X})\), any \(p \geq 1\) and every \(t = 1, 2, ..., T\), we have the inequality

\[
\|(f, \rho_t^N) - (f, \rho_t)\|_p \leq \frac{\tilde{c}_t \|f\|_{\infty}}{\sqrt{N}},
\]

where \(\tilde{c}_t < \infty\) is a constant independent of \(N\).

**Proof:** For \(t = 0\), \(\rho_0^N = \pi_0^N\), hence the result follows from Lemma \((1)\). At any time \(t > 0\), since \(\rho_t^N = G_t^N \pi_t^N\), we readily have

\[
E \left[ \left\| (f, \rho_t^N) - (f, \rho_t) \right\|^p \right] = E \left[ \left\| \frac{1}{N} \sum_{i=1}^{N} G_t^N f(x_t^{(i)}) - (f, \rho_t) \right\|^p \right]
= E \left[ \left\| \frac{1}{N} \sum_{i=1}^{N} Z_t^{(i)} \right\|^p \right],
\]

where \(Z_t^{(i)} = G_t^N f(x_t^{(i)}) - (f, \rho_t)\), \(i = 1, ..., N\). It is apparent that the random variables \(Z_t^{(i)}\), \(i = 1, ..., N\), are conditionally independent given the \(\sigma\)-algebra \(\mathcal{F}_t\), generated by the set \(\{x_{0:t-1}^{(i)}, x_{0:t}^{(j)} : 1 \leq j \leq N\}\). It can also be proved that every \(Z_t^{(i)}\) is centered and bounded, as explicitly shown in the sequel.

To see that \(Z_t^{(i)}\) has zero mean, let us note first that

\[
E \left[ G_t^N f(x_t^{(i)}) | \mathcal{F}_t \right] = G_t^N (f, \pi_t^N),
\]

since \(G_t^N\) is measurable w.r.t. \(\mathcal{F}_t\). Moreover, by the same argument as in the proof of Lemma \((2)\) one can show that \(G_t^N (f, \pi_t^N) = G_{t-1}^N (f g_t^{\pi_t^N}, \xi_{t-1}^N)\) and, therefore,

\[
E \left[ G_t^N f(x_t^{(i)}) | \mathcal{F}_{t-1} \right] = E \left[ G_{t-1}^N (f g_t^{\pi_t^N}, \xi_{t-1}^N) | \mathcal{F}_{t-1} \right] = G_{t-1}^N ((f g_t^{\pi_t}, \tau_t), \pi_{t-1}^N),
\]

where we have used the fact that, for any \(h \in B(\mathcal{X})\), \(E((h, \xi_{t-1}^N) | \mathcal{F}_{t-1}) = (\langle h, \tau_t \rangle, \pi_{t-1}^N)\). However, since \(\rho_{t-1}^N = G_{t-1}^N \pi_{t-1}^N\), Eq. \((20)\) amounts to

\[
E \left[ G_t^N f(x_t^{(i)}) | \mathcal{F}_{t-1} \right] = ((f g_t^{\pi_t}, \tau_t), \rho_{t-1}^N)
\]

and taking (unconditional) expectations on both sides of the equation above yields

\[
E \left[ G_t^N f(x_t^{(i)}) \right] = E \left[ ((f g_t^{\pi_t}, \tau_t), \rho_{t-1}^N) \right]
= ((f g_t^{\pi_t}, \tau_t), \rho_{t-1})
= (f, \rho_t),
\]

\[\Box\]
where (27) follows from Lemma 2 (i.e., \( \rho_{t-1}^N \) is unbiased) and (28) is a straightforward consequence of the definition of \( \rho_t \) in (7). Equation (28) states that 
\[ E[Z_t^{(i)}] = E[G_t^N f(x_t^{(i)}) - (f, \rho_t)] = 0. \]

To see that (every) \( Z_t^{(i)} \) is bounded, note that, for any finite \( t \),
\[ G_t^N \leq \prod_{k=1}^{t} \| g_k^{y_{tk}} \|_\infty < \infty, \]
whereas
\[
(f, \rho_t) = ((f g_t^{y_{tk}}, \tau_t), \rho_{t-1})
= (((f g_t^{y_{tk}}, \tau_t) g_{t-1}^{y_{t-1}}, \tau_{t-1}) g_{t-2}^{y_{t-2}}, ..., \tau_1), \pi_0)
\leq \| f \|_\infty \prod_{k=1}^{t} \| g_k^{y_{tk}} \|_\infty < \infty. \] (30)

Taking (29) and (30) together we arrive at
\[ |Z_t^{(i)}| \leq 2 \| f \|_\infty \prod_{k=1}^{t} \| g_k^{y_{tk}} \|_\infty \] (31)
which is finite for any finite \( t \) (indeed, for every \( t \leq T \)).

Since the variables \( Z_t^{(i)}, i = 1, ..., N, \) in (26) are bounded, with zero mean and conditionally independent given \( \bar{\mathcal{F}}_t \), it is not difficult to show (see, e.g., [6, Lemma A.1]) that
\[ E \left[ |(f, \pi_t^N) - (f, \rho_t)|^p \right] \leq 2 \hat{c}_t^p \| f \|_\infty^p \prod_{k=1}^{t} \| g_k^{y_{tk}} \|_\infty^p, \] (32)
where the constant \( \hat{c}_t \) is finite and independent of \( N \). From (32) we easily obtain the inequality \( \| f \|_\infty \prod_{k=1}^{t} \| g_k^{y_{tk}} \|_\infty < \infty \) for any \( t \leq T < \infty \).

Finally, Lemmas 2 and 3 together enable us to compute explicit rates for the bias in the approximation of \( (f, \pi_t) \). To be specific, we can prove the following theorem.

**Theorem 1** If \( 0 < (1, \rho_t) < \infty \) for \( t = 1, 2, ..., T \) and Assumptions 1 and 2 hold, then, for any \( f \in B(X) \) and every \( 0 \leq t \leq T \), we obtain
\[ |E[(f, \pi_t^N) - (f, \pi_t)]| \leq \frac{\hat{c}_t \| f \|_\infty}{N}, \]
where \( \hat{c}_t < \infty \) is a constant independent of \( N \).

**Proof:** Let us first note that \( (f, \pi_t) = (f, \rho_t)/(1, \rho_t) \) and
\[
(f, \pi_t^N) = \frac{(f, \rho_t^N)}{G_t^N} \] (33)
\[ = \frac{(f, \rho_t^N)}{G_t^N} \frac{1}{(1, \pi_t^N)} \] (34)
\[ = \frac{(f, \rho_t^N)}{(1, \rho_t^N)}, \] (35)

11
where (33) follows from the construction of \( \rho_t^N \), (34) holds because \( (1, \pi_t^N) = 1 \) and (35) is, again, a consequence of the construction of \( \rho_t^N \). Therefore, the difference \( (f, \pi_t^N) - (f, \pi_t) \) can be written as

\[
(f, \pi_t^N) - (f, \pi_t) = \frac{(f, \rho_t^N)}{(1, \rho_t^N)} - \frac{(f, \rho_t)}{(1, \rho_t)}
\]

and, since \((f, \rho_t) = E[(f, \rho_t^N)]\) (from Lemma 2), the bias can be expressed as

\[
E[(f, \pi_t^N) - (f, \pi_t)] = E \left[ \frac{(f, \rho_t^N)}{(1, \rho_t^N)} - \frac{(f, \rho_t)}{(1, \rho_t)} \right].
\]  (36)

Some elementary manipulations on (36) yield the equality

\[
E[(f, \pi_t^N) - (f, \pi_t)] = E \left[ (f, \pi_t^N) \frac{(1, \rho_t)}{(1, \rho_t^N)} - \frac{(f, \rho_t)}{(1, \rho_t)} \right].
\]  (37)

If we realize that \( E[(1, \rho_t) - (1, \rho_t^N)] = 0 \) (again, a consequence of Lemma 2) and move the factor \((1, \rho_t)^{-1}\) out of the expectation, then we easily rewrite Eq. (37) as

\[
E[(f, \pi_t^N) - (f, \pi_t)] = \frac{1}{(1, \rho_t)} E \left[ (f, \pi_t^N) \left( (1, \rho_t) - (1, \rho_t^N) \right) \right] - \frac{(f, \pi_t)}{(1, \rho_t)} E \left[ (1, \rho_t) - (1, \rho_t^N) \right]
\]

\[
= \frac{1}{(1, \rho_t)} E \left[ ((f, \pi_t^N) - (f, \pi_t)) \left( (1, \rho_t) - (1, \rho_t^N) \right) \right]
\]

\[
\leq \frac{1}{(1, \rho_t)} \left( E \left[ (f, \pi_t^N)^2 \right] E \left[ (f, \pi_t)^2 \right] \right)^{1/2}
\]

\[
\leq \frac{1}{(1, \rho_t)} \left( \frac{c_t \|f\|_{\infty}^2}{N} \times \frac{\hat{c}_t}{N} \right)
\]

\[
= \frac{c_t \hat{c}_t \|f\|_{\infty}^2}{N} < \infty
\]  (38)\]

where we have applied the Cauchy-Schwartz inequality to obtain (38), (39) follows from Lemmas 1 and 2 and the constant

\[
\hat{c}_t = \frac{c_t \hat{c}_t \|f\|_{\infty}^2}{(1, \rho_t)} < \infty
\]

is independent of \( N \).

\( \square \)

For any \( f \in B(\mathcal{X}) \), let \( \mathcal{E}_t^N(f) \) denote the approximation difference, i.e.,

\[
\mathcal{E}_t^N(f) \triangleq (f, \pi_t^N) - (f, \pi_t).
\]

This is a random variable whose second order moment yields the MSE of \((f, \pi_t^N)\). It is straightforward to obtain a bound for the MSE from Lemma 1 and, by subsequently using Theorem 1, one also readily finds a similar bound for the variance of \( \mathcal{E}_t^N(f) \), denoted \( \text{Var}[\mathcal{E}_t^N(f)] \). These results are explicitly stated by the corollary below.

**Corollary 1** If \( 0 < (1, \rho_t) < \infty \) for \( t = 1, 2, ..., T \) and Assumptions 1 and 2 hold, then, for any \( f \in B(\mathcal{X}) \) and any \( 0 \leq t \leq T \), we obtain

\[
E \left[ \left( \mathcal{E}_t^N(f) \right)^2 \right] \leq \frac{c_t^2 \|f\|_{\infty}^2}{N} \quad \text{and}
\]

\[
\text{Var}[\mathcal{E}_t^N(f)] \leq \frac{(c_t^2 \|f\|_{\infty}^2)}{N},
\]  (40)\]

(41)
where \( c_t \) and \( c^t \) are finite constants independent of \( N \).

**Proof:** The inequality (40) for the MSE is a straightforward consequence of Lemma 1. Moreover, we can write the MSE in terms of the variance and the bias as

\[
E \left[ (\mathcal{E}^N_t(f))^2 \right] = \text{Var} [\mathcal{E}^N_t(f)] + E^2 [\mathcal{E}^N_t] \leq \frac{c^2_t \|f\|_\infty^2}{N}.
\]

(42)

Since Theorem 1 yields \( |E[\mathcal{E}^N_t]| \leq \hat{c}_t \|f\|_\infty \), then the inequality (42) implies that there exists a constant \( c^t \) such that (41) holds. \( \Box \)

### 3.3 Error rate for the ensemble approximation

Let us run \( M \) independent particle filters with the same (fixed) sequence of observations \( Y_{1:T} = y_{1:T}, T < \infty, \) and \( N \) particles each. The approximate measures output by the \( m \)-th filter are denoted \( \xi^m_{t,N}, \pi^m_{t,N} \) and \( \rho^m_{t,N} \), respectively, with \( m = 1, 2, \ldots, M \). Obviously, all the theoretical properties established in Section 3.2, as well as the basic Lemma 1, hold for each one of the \( M \) independent filters.

**Definition 1** The ensemble approximation of \( \pi_t \) with \( M \) independent filters is the discrete random measure \( \pi^{M \times N}_t \) constructed as the average

\[
\pi^{M \times N}_t = \frac{1}{M} \sum_{m=1}^{M} \pi^m_{t,N}.
\]

Obviously, similar ensemble approximations can be given for \( \xi_t \) and \( \rho_t \). Moreover, the statistical independence of the particle filters yields the following corollary as a straightforward consequence of Theorem 1 and Corollary 1.

**Corollary 2** If \( 0 < (1, \rho_t) < \infty \) for \( t = 1, 2, \ldots, T \) and Assumptions 1 and 2 hold, then, for any \( f \in B(\mathcal{X}) \) and any \( 0 \leq t \leq T \), the inequality

\[
E \left[ ((f, \pi^{M \times N}_t) - (f, \pi_t))^2 \right] \leq \frac{(c^t)^2 \|f\|_\infty^2}{MN} + \frac{\hat{c}^2_t \|f\|_\infty^2}{N^2}
\]

holds for some constants \( c^t \) and \( \hat{c}_t \) independent of \( N \) and \( M \).

**Proof:** Let us denote

\[
\mathcal{E}^{M \times N}_t(f) = (f, \pi^{M \times N}_t) - (f, \pi_t) \quad \text{and} \quad \mathcal{E}^{m,N}_t(f) = (f, \pi^{m,N}_t) - (f, \pi_t)
\]

for \( m = 1, 2, \ldots, M \). Since \( \pi^{M \times N}_t \) is a linear combination of i.i.d. random measures, it is apparent that

\[
E \left[ \mathcal{E}^{M \times N}_t(f) \right] = E \left[ \mathcal{E}^{m,N}_t(f) \right] \leq \frac{\hat{c}_t \|f\|_\infty}{N}, \quad \text{for any } m \leq M,
\]

(44)

where the inequality follows from Theorem 1. Moreover, because of the independence, we readily obtain

\[
\text{Var} \left[ \mathcal{E}^{M \times N}_t(f) \right] = \frac{1}{M} \text{Var} \left[ \mathcal{E}^{m,N}_t(f) \right] \leq \frac{(c^t)^2 \|f\|_\infty^2}{MN},
\]

(45)
where the inequality follows from Corollary 1. Since \( E[(\mathcal{E}_t^{M \times N})^2] = \text{Var}[\mathcal{E}_t^{M \times N}] + E^2[\mathcal{E}_t^{M \times N}] \), combining (43) and (44) yields (43) and concludes the proof. \( \square \)

The inequality in Corollary 2 admits several interpretations. Assume that some “computational budget” is given, i.e., that we have the resources to generate and update over time a fixed number \( K \) of particles. If we run Algorithm 1 with \( K \) particles, then, according to Lemma 1, the MSE \( E[(\mathcal{E}_t^K)^2] \) vanishes asymptotically with rate \( O(1/K) \). The same result is obtained via Corollary 2: a centralized particle filter with \( K \) particles is the degenerate case of the ensemble approximation in Definition 1 with \( M = 1 \) and \( N = K \), hence Corollary 2 also states that the MSE decreases with rate \( O(1/K) \). However, Corollary 2 also yields the same MSE rate when a proper ensemble is constructed, with some \( M > 1 \) and \( N = K/M \). Indeed, we can state the following Remark.

Remark 4 If the total number of particles \( K \) is fixed, then the ensemble approximation \( \pi_{M \times N}^{M \times N} \) with \( K = MN \) and the standard (bootstrap filter) approximation with \( K \) particles, \( \pi_K^K \), yield the same MSE rate \( O(1/K) \). Since the ensemble approximation is obtained by averaging over \( M \) independent particle filters, the interactions among particles are strictly constrained to be local (within the \( M \) subsets of size \( N \) assigned to the independent particle filters). This implies that if the \( M \) particle filters are run in parallel, there is no parallelization overhead in terms of interaction among the parallel processing units.

Remark 5 According to the inequality (43), the error rate is \( O\left(\frac{1}{MN}\right) \) as long as \( N \geq M \). Otherwise, the term \( \frac{\hat{c}^2\|f\|_2^2}{N^2} \) becomes dominant and the resulting asymptotic error bound turns out higher.

3.4 Comparison of parallelization schemes via time–error indices

Obviously, the expected advantage of parallel computation is the drastic reduction of the time needed to run the particle filter. Let the running time for a particle filter with \( K \) particles be of order \( T(K) \), where \( T : N \to (0, \infty) \) is some strictly increasing function of \( K \). The quantity \( T(K) \) includes the time needed to generate new particles, weight them and perform resampling. The latter step is the bottleneck for parallelization, as it requires the interaction of all \( K \) particles. Also, a “straightforward” implementation of the resampling step leads to an execution time \( O(K \log(K)) \), although efficient algorithms exist that achieve to a linear time complexity, \( T(K) = K \). We can combine the MSE rate and the time complexity to propose a a time–error performance metric.

Definition 2 We define the time–error index of a particle filtering algorithm with running time of order \( T \) and asymptotic MSE rate \( R \) as \( C \triangleq T \times R \).

For the standard (centralized) particle filter with \( K \) particles, the time complexity is denoted \( T(K) \) and the MSE rate is of order \( K^{-1} \), hence the time–error index becomes

\[ C_{spf}(K) = \frac{T(K)}{K} \]
If the implementation is good enough to attain a linear time complexity, $T(K) = K$, then the time–error index turns out to be constant, namely $C_{spf} = 1$.

For the computation of the ensemble approximation $\pi_t^{M\times N}$ we can run $M$ independent particle filters in parallel, with $N = K/M$ particles each and no interaction among them, and hence the execution time becomes of order $T(N)$, with $T(N) = N$ if an appropriate implementation is ensured. Since the error rate for the ensemble approximation is of order $O\left(\frac{1}{MN} + \frac{1}{N^2}\right)$, the time–error index of the ensemble approximation is

$$C_{ipf}(M, N) = T(N) \left(\frac{1}{MN} + \frac{1}{N^2}\right).$$

As before, if the implementation is good enough to yield $T(N) = N$, then the time–error index becomes

$$C_{ipf} = \frac{1}{M} + \frac{1}{N}$$

and hence it vanishes with $M, N \to \infty$. In particular, since we have to choose $N \geq M$ to ensure a rate of order $\frac{1}{MN}$, we can choose $N = N(M) \geq M$ and then $\lim_{M \to \infty} C_{ipf} = 0$. In any case, whenever $N > 1$ it is apparent that $C_{ipf} < C_{spf}$.

### 3.5 Expectation of the approximate filter

Besides the computational view of Sections 3.3 and 3.4, Theorem 1 can also be exploited to assess the ensemble approximation $\pi_t^{M\times N}$ in terms of the total variation distance defined as follows.

**Definition 3** Let $\alpha, \beta \in \mathcal{P}(\mathcal{X})$ be two probability measures. The total variation distance (TVD) between $\alpha$ and $\beta$ is usually defined as

$$d_{TV}(\alpha, \beta) \triangleq \sup_{A \in \mathcal{B}(\mathcal{X})} |(I_A, \alpha) - (I_A, \beta)|,$$

where $I_A$ is the indicator function

$$I_A(x) = \begin{cases} 1, & \text{if } x \in A, \\ 0, & \text{otherwise.} \end{cases}$$

The ensemble $\pi_t^{M\times N} = \frac{1}{M} \sum_{m=1}^M \pi_t^{m,N}$ can be interpreted as an empirical approximation of the expectation of the measured-valued random variable $\pi_t^N$. Indeed, if we introduce

$$\hat{\pi}_t^N \triangleq E[\pi_t^N]$$

then it is apparent that, for any $f \in \mathcal{B}(\mathcal{X})$,

$$E \left[ (f, \pi_t^N) - (f, \pi_t^N) \right] = (f, \hat{\pi}_t^N) - (f, \pi_t),$$

and $(f, \pi_t^{M\times N}) = \frac{1}{M} \sum_{m=1}^M (f, \pi_t^{m,N})$ is the sample-mean approximation of $(f, \hat{\pi}_t^N)$. Theorem 1 can be restated immediately as follows.
Corollary 3 If \( 0 < (1, \rho_t) < \infty \) for \( t = 1, 2, ..., T \) and Assumptions \( \boxed{1} \) and \( \boxed{5} \) hold, then, for every \( 0 \leq t \leq T \), we obtain that
\[
\text{dTV} \left( \hat{\pi}_t^N, \pi_t \right) \leq \frac{\hat{c}_t}{N},
\]
where \( \hat{c}_t < \infty \) is a constant independent of \( N \).

Proof: From the definition of total variation distance,
\[
\text{dTV} \left( \hat{\pi}_t^N, \pi_t \right) = \sup_{A \in B(X)} \left| (I_A, \hat{\pi}_t^N) - (I_A, \pi_t) \right| \leq \sup_{f \in B(X), \|f\|_{\infty} \leq 1} \left| (f, \hat{\pi}_t^N) - (f, \pi_t) \right|,
\]
since \( I_A \in B(X) \) and \( \|I_A\|_{\infty} = 1 \) for every Borel set \( A \). However, \( (f, \hat{\pi}_t^N) - (f, \pi_t) = E \left[ (f, \pi_t^N) - (f, \pi_t) \right] \), hence a straightforward application of Theorem \( \boxed{1} \) completes the proof. \( \square \)

4 Numerical example

4.1 The 3-dimensional Lorenz system

Let us show some simple computer simulation results in order to illustrate the numerical performance of the proposed independent parallelization scheme. For this purpose, we consider the problem of tracking the proposed independent parallelization scheme. For this purpose, we consider the problem of tracking the state of a 3-dimensional Lorenz system \( \boxed{17} \) with additive dynamical noise and partial observations \( \boxed{4} \). To be specific, consider a 3-dimensional stochastic process \( \{X(s)\}_{s \in (0, \infty)} \) taking values on \( \mathbb{R}^3 \), whose dynamics is described by the system of stochastic differential equations
\[
\begin{align*}
\text{d}X_1 &= -s(X_1 - Y_1) \, dW_1, \\
\text{d}X_2 &= rX_1 - X_2 - X_1X_3 \, dW_2, \\
\text{d}X_3 &= X_1X_2 - bX_3 \, dW_3,
\end{align*}
\]
where \( \{W_i(s)\}_{s \in (0, \infty)}, i = 1, 2, 3 \), are independent 1-dimensional Wiener processes and
\[
(s, r, b) = \left( 10, 28, \frac{8}{3} \right)
\]
are static model parameters (which yield chaotic dynamics). A discrete-time version of the latter system using Euler’s method with integration step \( \Delta = 10^{-3} \) is straightforward to obtain and yields the model
\[
\begin{align*}
X_{1,n} &= X_{1,n-1} - \Delta s(X_{1,n-1} - X_{2,n-1}) + \sqrt{\Delta} U_{1,n}, \\
X_{2,n} &= X_{2,n-1} + \Delta (rX_{1,n-1} - X_{2,n-1} - X_{1,n-1}X_{3,n-1}) + \sqrt{\Delta} U_{2,n}, \\
X_{3,n} &= X_{3,n-1} + \Delta (X_{1,n-1}X_{2,n-1} - bX_{3,n-1}) + \sqrt{\Delta} U_{3,n},
\end{align*}
\]
(47)–(49)

where \( \{U_{i,n}\}_{n=0,1,...}, i = 1, 2, 3 \), are independent sequences of i.i.d. normal random variables with 0 mean and variance 1. System \( \boxed{17} \)–\( \boxed{19} \) is partially observed every 100 discrete-time steps. Specifically, we collect a sequence of scalar observations \( \{Y_t\}_{t=1,2,...} \), of the form
\[
Y_t = X_{1,100t} + V_t,
\]
(50)

where
\[
V_t = X_{2,100t} + \sqrt{\Delta} U_{2,100t},
\]
(51)
where \( \{ V_t\}_{t=1,2,...} \) is a sequence of i.i.d. normal random variables with zero mean and variance \( \sigma^2 = \frac{1}{2} \).

Let \( X_n = (X_{1,n}, X_{2,n}, X_{3,n}) \in \mathbb{R}^3 \) be the state vector. The dynamic model given by Eqs. (47)–(49) yields the family of kernels \( \tau_{n,\theta}(dx|x_{n-1}) \) and the observation model of Eq. (50) yields the likelihood function

\[
g_{t,\theta}^{y_t}(x_{100t}) \propto \exp\left\{ -\frac{1}{\sigma^2}(y_t - x_{1,100t}) \right\},
\]

both in a straightforward manner. The goal is to track the sequence of joint posterior probability measures \( \pi_t, t = 1,2,... \), for \( \{ \hat{X}_t \}_{t=1,...} \), where \( \hat{X}_t = \hat{x}_t \) conditional on \( \hat{X}_{t-1} = \hat{x}_{t-1} \) by successively simulating

\[
\tilde{x}_n \sim \tau_{n,\theta}(dx|\tilde{x}_{n-1}), \quad n = 100(t-1) + 1,...,100t,
\]

where \( \tilde{x}_{100(t-1)} = \hat{x}_{t-1} \) and \( \hat{x}_t = \tilde{x}_{100t} \). The prior measure for the state variables is normal, namely

\[
X_0 \sim \mathcal{N}(x_*, v_0^2 I_3),
\]

where \( x_* = (-10.2410; -1.3984; -23.6752) \) is the mean and \( v_0^2 I_3 \) is the covariance matrix, with \( v_0^2 = 10 \) and \( I_3 \) the 3-dimensional identity matrix.

### 4.2 Simulation setup

In this section we aim at illustrating the gain in relative performance, taking into account both estimation error and running time, that can be attained using the proposed scheme for parallelization using independent filters. With this purpose, we have applied

- the standard bootstrap filter (Algorithm 1), termed BF in the sequel,
- the island particle method of [22], and
- the ensemble of independent bootstrap filters that we have investigated in Section 3

to the tracking of the sequence of probability measures \( \pi_t \) generated by the 3-dimensional Lorenz model described in Section 4.1. We have generated a sequence of 200 artificial observations, \( \{ y_t; t = 1,...,200 \} \), spread over an interval of 20 seconds (in continuous time), corresponding to \( 2 \times 10^4 \) discrete time steps in the Euler scheme (hence, one observation every 100 steps). The same observation sequence has been used for all the simulations.

The ensemble of independent particle filters consists of \( M \) filters with \( N \) particles each, abiding by the notation in Section 3 with resampling for every \( t = 1,2,... \), i.e., every time an observation is collected and processed to obtain importance weights. Since the time scale of the Euler approximation of Eqs. (47)–(49) is \( n = 100t \), a resampling step is taken every 100 steps of the underlying discrete-time system. Similarly, the island particle algorithm consists of \( M \) subsets of particles (islands, in the terminology of [22]) with \( N \) particles per subset. Within each island, the \( N \) particles are resampled for every \( t = 1,2,...,\)
while the islands are resampled for $t = 5k$, $k = 1, 2, \ldots$. The intra-island resampling can be carried out in parallel for the $M$ islands, while the island resampling does not admit parallelization. The standard bootstrap filter runs with $K$ particles, where $K = MN$ (unless explicitly stated) for a fair comparison, and resampling for $t = 1, 2, \ldots$.

We have coded the three algorithms in matlab (version 7.11.0.584 [R2010b]) and run the experiments using a pool of multi-processor machines, each one having 8 cores at 3.16 GHz and 32 GB of RAM memory. The standard (centralized) BF is run with $K = NM$ particles in a single core. For the ensemble of independent filters, each BF with $N$ particles is run separately and independently in one core, hence taking $M$ cores overall. In the case of the island particle method we have run the algorithm in a purely sequential manner, i.e., we have not truly parallelized the intra-resampling steps. Instead, to calculate comparable running times, for each $t$ we have recorded the time taken by each one of the $M$ intra-resampling steps and then computed their average. These averages are used for the calculation of running times instead of the actual times.

To assess the approximation errors, we have computed empirical MSEs for the approximation of the posterior mean, $E[\hat{X}_t|Y_{1:t}] = (1, \pi_t)$, for the three algorithms at the last update step, $t = 200$. Note, however, that the integral $(1, \pi_t)$ cannot be computed in closed form for this system. Therefore, we have used the “expensive” estimate

$$(1, \pi_t) \approx (1, \pi_t^J), \quad \text{with } J = 10^5 \text{ particles},$$

computed via the standard BF, as a proxy of the true value.

### 4.3 Numerical results

Figure 1 (left) displays the empirical MSE, averaged over 100 independent simulation runs, attained by the parallel schemes when the number of filters (respectively, islands) is fixed, $M = 20$, and the number of particles per filter (island) ranges from $N = 100$ to $N = 1000$. The outcome of the centralized BF with $K = MN$ particles, hence ranging from $K = 20 \times 100$ to $K = 20 \times 1000$, is also shown for comparison.

We observe that proposed ensemble of independent bootstrap filters achieves a poor performance when the number of particles per filter, $N$, is relatively low ($N = 100$), while for moderate values ($N \geq 400$) it nearly matches the MSE of the centralized BF. The island particle method is more accurate than the independent ensemble for $N = 100, 200$, as it takes advantage of the interaction among the islands, but displays a slightly worse MSE than the centralized BF and the independent ensemble for $N \geq 400$.

The empirical variance of the MSE for the same set of 100 simulation trials is displayed in Figure 1 (right). The results show, again, that the island BF algorithm makes an efficient use of the island resampling step when $N$ is low, so that the $M$ islands remain balanced and the overall filter works properly, but falls short of the independent ensemble scheme for larger values of $N$. 

18
Figure 1: Empirical mean (left) and variance (right) of the MSE for the centralized BF, the independent ensemble of BFs and the island particle method with $M = 20$ constant and $N = 100, 200, 400, 800, 1000$. All curves have been obtained from a set of 100 independent simulation trials.

Figure 2 shows the running time for the independent ensemble and the island particle method when the number of particles per filter (island) $N = 500$ is held constant and the number of filters (islands) ranges from $M = 2$ to $M = 200$. The running time for a centralized filter with $K = 500M$ particles is also shown for comparison. It is seen that the average (over 10 simulations) running time for the independent ensemble scheme is constant, as it only depends on $N$, while the running time for the centralized algorithm grows quickly with $M$. It is remarkable that the island particle method, in the configuration chosen for these simulations (one island-level resampling every 5 observations processed) does not bring any gain in running time: the gain achieved by parallelizing the intra-island resampling is compensated for by the overhead of the island-level resampling steps.

Finally, we look into the relationship between the MSE and the running time for the three algorithms. With the number of filters (islands) $M = 20$ fixed, we have run 100 independent simulation trials for each value $N = 100, 200, 400, 800$ and $1000$, and computed the empirical MSE and the average running time for the two parallel schemes and each combination of $M$ and $N$. Correspondingly, we have also run the centralized BF with $K = MN$ particles, hence for $K = 2 \times 10^3, 4 \times 10^3, 8 \times 10^3, 16 \times 10^3$ and $20 \times 10^3$.

Figure 3 (left) displays the resulting empirical MSE versus the running time for the three methods. If we qualify an algorithm as more efficient than another one when it is capable of attaining a lower MSE in the same amount of time, then this set of simulations shows that the independent ensemble scheme is more efficient than both the centralized BF and the island particle method. Indeed, a close look at Figure 3 (left) reveals that the ensemble of $M = 20$ independent BFs with $N = 1000$ particles per filter achieves an empirical MSE of $\approx 6 \times 10^{-4}$ with a running time of $\approx 2.9$ seconds, while the centralized BF attains the same performance with $K = 20 \times 800$ particles and a running time of $\approx 27.2$ seconds (as shown by the
Figure 2: Running time versus the number of filters (islands) for the independent ensemble and island particle schemes, when the number of particles per filter (island) is kept constant, \( N = 500 \). The running times of the centralized BF correspond to a total number of particles of the form \( K = 500M \), with \( M = 2, \ldots, 200 \).

dashed horizontal line in the plot). The island particle method falls short of this MSE value even with \( M = 20 \) and \( N = 1000 \). Figure 3 (right) shows the empirical variance of the MSE, versus the running time, for the same set of computer simulations.

5 Discussion

We have addressed the problem of parallelizing the standard particle filtering algorithm by splitting the total number of particles \( K \) into \( M \) subsets, running one independent particle filter per subset, and then building an average filter measure using the ensemble of i.i.d. random approximations produced by the filters. This approach avoids all dependences among the filters, an idea which goes against recent approaches to the problem \cite{2, 23, 19, 22} and, to some extent, against the intuition that a certain interaction is needed to make the \( M \) filters, with \( N \) particle each, work together with the same performance as a centralized filter with \( K = MN \) particles.

The rationale to advocate the averaging of independent particle filters instead of the (algorithmically more sophisticated) schemes based on controlled interactions, such as \cite{2, 23, 22}, relies on both theoretical and practical arguments. Theoretically, the simple analysis in this paper shows that the parallelization based on independent filters has the same asymptotic performance as a centralized particle filter. We have obtained this result by looking at the mean square approximation error for integrals of bounded

\footnote{While we have restricted the research to the standard (bootstrap) filter for simplicity of presentation, the analysis, and the whole argument about parallelization by means of independent filters, extends in an almost straightforward manner to more sophisticated algorithms that may use tailored importance functions for the generation of particles.}
Figure 3: Empirical mean (left) and variance (right) of the MSE versus the running time for the centralized BF, the independent ensemble of BFs and the island particle method. The two parallel schemes are run with $M = 20$ constant and $N = 100, 200, 400, 600, 800$ and 1000. The centralized BF is run with $K = 20N$ particles, where $N$ takes values in the same way as for the parallel algorithms. The dashed horizontal lines indicate where the mean (left) and variance (right) of the MSE match for the independent ensemble and the centralized BF. The running times for the two algorithms at that MSE level are shown as labels on the horizontal axis.

functions w.r.t. the filter measure, decomposed in variance and bias terms. The bounds we have obtained depend explicitly on $M$ (the number of filters) and $N$ (the number of particles per filter), and show that there is no asymptotic performance loss for schemes with $N \geq M$. This result is actually aligned with recent contributions in the field of machine learning regarding the statistical properties of averages of independent estimators (including classifiers and regressors) [21]. We have also utilized the asymptotic convergence rates to propose a time–error index that enables a quantitative comparison of centralized and parallel particle filtering scheme in terms of asymptotic accuracy and running time. These analytical results can be extended to account for stronger forms of convergence (under additional assumptions on the model) and adapted to continuous-time state-space systems (see [14] for the analysis of the bias and the MSE in this context).

From a practical perspective, we have shown that the averaging of independent filters should be preferred when $N$, the number of particles per independent filter, is sufficiently large. Indeed, our computer simulations suggest that if we seek a scheme with a large number of parallel filters ($M$) and a relatively small number of particles per filter ($N$) then parallelization schemes that exploit a certain degree of interaction between filters should be preferred—at the expense of a computational overhead to implement such interaction. On the other hand, if $N$ is large enough to make the parallel filters work (even roughly), then our simulations show that interaction is not needed anymore and independence can be fully exploited both in terms of accuracy and running time.
The interest in designing particle filtering schemes that can have fast implementations using parallel hardware obeys the surge of several problems in science (meteorology, oceanography, climate science, biochemistry or systems medicine) and engineering (sensor networks, multi target tracking) where the fundamental task is the tracking of a complex (e.g., high dimensional and/or chaotic) dynamical system. In this paper, we have complemented the theoretical analysis with computer simulation results for a Lorenz 65 chaotic system with dynamical noise (modeled as a 3-dimensional Wiener process) and partial observations. The numerical outcomes of the simulations show that the proposed scheme, consisting of an ensemble of independent particle filters, is more efficient than the corresponding centralized algorithm (meaning that it can attain a given estimation error with a lesser running time) in a high-performance computing system consisting of a pool of multi-core CPUs. For a further comparison, we have also included in the computer simulation study a recently proposed parallelization scheme [22] that relies on two levels of resampling. Our results show that the overhead associated to this scheme can make it inefficient (possibly depending on the choice of parameters for the algorithm) compared even to the centralized standard particle filter. We interpret this fact as an indication that the independent ensemble scheme can be, because of its simplicity and sound asymptotic behavior, the preferred choice for many practical applications even when compared with more sophisticated parallelization techniques.

Acknowledgements

The work of J. Míguez was partially supported by Ministerio de Economía y Competitividad of Spain (program Consolider-Ingenio 2010 CSD2008-00010 COMONSENS and project TEC2012-38883-C02-01 COMPREHENSION) and Ministerio de Educación, Cultura y Deporte of Spain (Programa Nacional de Movilidad de Recursos Humanos PRX12/00690). D. C. and J. M. would also like to acknowledge the support of the Isaac Newton Institute through the program “Monte Carlo Inference for High-Dimensional Statistical Models”.

References

[1] A. Bain and D. Crisan. Fundamentals of Stochastic Filtering. Springer, 2008.
[2] M. Bolić, P. M. Djurić, and S. Hong. Resampling algorithms and architectures for distributed particle filters. IEEE Transactions Signal Processing, 53(7):2442–2450, July 2005.
[3] O. Cappé, S. J. Godsill, and E. Moulines. An overview of existing methods and recent advances in sequential Monte Carlo. Proceedings of the IEEE, 95(5):899–924, 2007.
[4] A. J. Chorin and P. Krause. Dimensional reduction for a Bayesian filter. PNAS, 101(42):15013–15017, October 2004.
[5] D. Crisan. Particle filters - a theoretical perspective. In A. Doucet, N. de Freitas, and N. Gordon, editors, Sequential Monte Carlo Methods in Practice, chapter 2, pages 17–42. Springer, 2001.

[6] D. Crisan and J. Miguez. Particle-kernel estimation of the filter density in state-space models. Bernoulli, (in press) (arXiv:1111.5866v5 [stat.CO]), 2013.

[7] P. Del Moral and A. Guionnet. On the stability of interacting processes with applications to filtering and genetic algorithms. Annales de l’Institut Henri Poincaré (B) Probability and Statistics, 37(2):155–194, 2001.

[8] R. Douc, O. Cappé, and E. Moulines. Comparison of resampling schemes for particle filtering. In Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis, pages 64–69, September 2005.

[9] A. Doucet, N. de Freitas, and N. Gordon. An introduction to sequential Monte Carlo methods. In A. Doucet, N. de Freitas, and N. Gordon, editors, Sequential Monte Carlo Methods in Practice, chapter 1, pages 4–14. Springer, 2001.

[10] A. Doucet, N. de Freitas, and N. Gordon, editors. Sequential Monte Carlo Methods in Practice. Springer, New York (USA), 2001.

[11] A. Doucet, S. Godsill, and C. Andrieu. On sequential Monte Carlo Sampling methods for Bayesian filtering. Statistics and Computing, 10(3):197–208, 2000.

[12] A. Gelencsér-Horváth, G. Tornai, A. Horváth, and G. Cseréy. Fast, parallel implementation of particle filtering on the gpu architecture. EURASIP Journal on Advances in Signal Processing, 2013(1):1–16, 2013.

[13] N. Gordon, D. Salmond, and A. F. M. Smith. Novel approach to nonlinear and non-Gaussian Bayesian state estimation. IEE Proceedings-F, 140(2):107–113, 1993.

[14] W. Han. On the Numerical Solution of the Filtering Problem. Ph.D. Thesis. Department of Mathematics, Imperial College London, 2013.

[15] O. Hlinka, O. Sluciak, F. Hlawatsch, P. Djuric, and M. Rupp. Likelihood consensus and its application to distributed particle filtering. IEEE Transactions on Signal Processing, 60(8):4334–4349, 2012.

[16] H. R. Künsch. Particle filters. Bernoulli, 19(4):1391–1403, 2013.

[17] E. N. Lorenz. Deterministic nonperiodic flow. Journal of Atmospheric Sciences, 20(2):130–141, 1963.

[18] J. Míguez. Analysis of selection methods for cost-reference particle filtering with applications to maneuvering target tracking and dynamic optimization. Digital Signal Processing, 17:787–807, 2007.

[19] J. Míguez. On the uniform asymptotic convergence of a distributed particle filter. In Proceedings of the 8th IEEE Sensor Array and Multichannel Signal Processing Workshop. IEEE, June 2014.
[20] J. Míquez, D. Crisan, and P. M. Djurić. On the convergence of two sequential Monte Carlo methods for maximum a posteriori sequence estimation and stochastic global optimization. *Statistics and Computing*, 23(1):91–107, 2013.

[21] J. Rosenblatt and B. Nadler. On the optimality of averaging in distributed statistical learning. *arxiv:1407.2724 [stat.ML]*, 2014.

[22] C. Vergé, C. Dubarry, P. Del Moral, and E. Moulines. On parallel implementation of sequential monte carlo methods: the island particle model. *Statistics and Computing*, pages 1–18, 2013.

[23] N. Whiteley, A. Lee, and K. Heine. On the role of interaction in sequential monte carlo algorithms. *arXiv:1309.2918 [stat.CO]*, 2013.