PARALLELIZATION OF SEQUENTIAL MONTE CARLO METHODS USING PARTICLE ISLANDS: THE B^2ASIL ALGORITHM

C. Vergé*, E. Moulines†, J. Olsson‡

* ONERA — The French Aerospace Lab
  Palaiseau, France
† Telecom ParisTech/ CNRS UMR LTCI
  Paris, France
‡ Department of Mathematics
  KTH Royal Institute of Technology
  Stockholm, Sweden

ABSTRACT

Particle island methods [1, 2] implement particle filters in a parallel architecture by dividing the particle system into islands evolving according to local particle filters. The islands are however allowed to interact on the basis of their mean potential. In this note, focus is set on the double bootstrap algorithm with adaptive selection on the island level (B^2ASIL), in which island interaction is applied only sparingly according to a criterion measuring the skewness of the island weights. Besides discussing, using novel results convergence obtained in [3], the theoretical properties of B^2ASIL, we also illustrate the performance of the algorithm by simulations.

Index Terms — SMC methods, island models, particle filter, central limit theorem (CLT), parallelization.

1. INTRODUCTION

This paper discusses approaches to parallelization of sequential Monte Carlo methods (or particle filters) approximating normalized Feynman-Kac distribution flows. Interacting particle systems are used increasingly for sampling from complex high-dimensional distributions in a wide range of applications, including nonlinear filtering, data assimilation problems, rare event sampling, hidden Markov chain parameter estimation, stochastic control problems, and financial mathematics; see e.g. [4, 5] and the references therein. These algorithms evolve, recursively and randomly in time, a sample of random draws, particles, with associated importance weights, and the Feynman-Kac distribution flow is approximated by the weighted empirical measures associated with this evolving sample. The particle cloud is updated through selection and mutation operations, where the former duplicates or eliminates, through resampling, particles with large or small importance weights, respectively, while the latter disseminates randomly the particles over the state space and updates accordingly the importance weights for further selection.

Particle filtering is a computationally intensive method. Parallel implementation is an appealing solution to tackle this issue but is not straightforward due to the particle interaction caused by the selection operation. In [1] it is proposed to implement the interacting particle system in parallel as follows. Instead of considering a single large batch of \( N = N_1 N_2 \) particles, the population is divided into \( N_1 \) batches of \( N_2 \) particles. The batches are in the sequel referred to as islands. Each particle belonging to a given island will also be referred to as an individual. Each island evolves according to the usual selection and mutation operations. The islands may evolve, on different processors, independently or may interact through multinomial selection according to weights proportional to the weight averages over the different subpopulations. We will consider two kinds of island interaction: systematic
resampling or adaptive resampling based on the coefficient of variation criterion, leading to the double bootstrap algorithm (B^2) and the double bootstrap algorithm with adaptive selection on the island level (B^2ASIL) algorithms, respectively, and where B^2 can be viewed as a particular case of the B^2ASIL algorithm. A sequence of Monte Carlo estimators (2) is obtained by weighing up, using the island weights, the self-normalized empirical measures associated with the different particle islands.

The theoretical analysis of B^2ASIL-type algorithms is challenging due to the intricate dependence structure imposed by the island selection operation and the “double asymptotics” introduced by \( N_1 \) and \( N_2 \), resp. island number and island size. Nevertheless, even though the islands are allowed to interact through selection, any two individuals of the system should become more and more statistically independent as the number of islands as well as the size of the islands grow (cf. the propagation of chaos property of standard SMC methods [6]). Thus, we may expect a law of large numbers as the size of the islands grow (cf. the the theoretical analysis of B^2ASIL-type algorithms introduced in [1, Algorithm 3], where selection on the island level is executed on the basis of the coefficient of variation (CV; see [10]) given by CV^2(\( \Omega_N(i) \)) = N_1 \sum_{i=1}^{N_1} \Omega_N^2(i) - 1. The CV is in one-to-one correspondence with the efficient sample size (ESS, proposed in [11] and used in [1]). The B^2ASIL scheme is detailed in Algorithm 1. Denote by, for \( n \in \mathbb{N} \) and \( h \in F_b(\mathcal{X}) \),
\[
\eta_n h = \sum_{i=1}^{N_1} \check{\Omega}_N^{(n)}(i) \sum_{j=1}^{N_2} \check{\omega}_N^{(n)}(i, j) h(\xi_N^{(n)}(i, j))
\]
the estimators returned by the B^2ASIL algorithm.

4. CONVERGENCE RESULTS FOR THE B^2ASIL ALGORITHM

In the following, let \( N = N_1 N_2 \) be the total number of individuals of each archipelago. The following results, describing the convergence of B^2ASIL as \( N_1 \) and \( N_2 \) tend jointly to infinity, were obtained in [3] through single-step analyses of the different selection and mutation operations acting on the archipelagos.

3. THE B^2ASIL ALGORITHM

For each \( i \in [1, N_1] \), we let \( \{(\xi_N(i, j), \omega_N(i, j))\}_{j=1}^{N_2} \) be an island of particles or individuals (the \( \xi_N \)), with associated importance weights (the \( \omega_n \)). The particle weights are assumed to be positive and uniformly bounded. Each island is assigned a positive weight \( \Omega_N(i) \), and the set of weighted islands, i.e., \( \{(\Omega_N(i), \{(\xi_N(i, j), \omega_N(i, j))\}_{j=1}^{N_2})\}_{i=1}^{N_1} \), is called an archipelago. Set \( \Omega_N(i) = \omega_N(i) / \sum_{\ell=1}^{N_1} \Omega_N(\ell) \) and \( \omega_N(i, j) = \omega_N(i, j) / \sum_{\ell=1}^{N_2} \omega_N(i, \ell) \).

In this section, our aim is to form, online, a sequence of archipelagos targeting the Feynman-Kac flow \( \{\eta_n\}_{n\in\mathbb{N}} \) defined in (1) by subjecting the archipelagos to a number of elementary operations such as selection on the island level, selection on the individual level and mutation. For all \( p \in \mathbb{N} \), let \( R_p \) be a (normalized) transition kernel on \( \mathcal{X} \) such that \( Q_p(x, \cdot) \ll R_p(x, \cdot) \) for all \( x \in \mathcal{X} \), and denote the corresponding Radon-Nikodym derivatives by \( w_p(x, \tilde{x}) = dQ_p(x, \cdot) / dR_p(x, \cdot) \), \((x, \tilde{x}) \in \mathcal{X}^2 \). We will focus on the B^2ASIL algorithm introduced in [1, Algorithm 3], where selection on the island level is executed on the basis of the coefficient of variation (CV; see [10]) given by
\[
CV^2(\Omega_N(i)) = N_1 \sum_{i=1}^{N_1} \Omega_N^2(i) - 1.
\]
The CV is in one-to-one correspondence with the efficient sample size (ESS, proposed in [11] and used in [1]).
Algorithm 1: The B^2ASIL algorithm

/* Initialization */
for i ← 1 to N_1 do
  for j ← 1 to N_2 do
    ξ_N(i, j) ~ η_0;
    ξ_N^{(1)}(i, j) ~ R_0(ξ_N^{(0)}(i, j), ·);
    ω_N(i, j) ← w_0(ξ_N^{(0)}(i, j), ξ_N^{(1)}(i, j));
  end
  Ω_N(i) ← ∑_{j=1}^{N_2} Ω_N^{(1)}(i, j)/N_2;
end

/* Island selection */
for p ← 1 to n − 1 do
  if CV^2(\{Ω_N^{(p)}(i)\}_{i=1}^{N_1}) > τ then
    for i ← 1 to N_1 do
      I_N(i) ~ Mult(\{\tilde{Ω}_N^{(p)}(i')\}_{i'=1}^{N_1});
    end
  else
    for i ← 1 to N_1 do
      I_N(i) ← i;
    end
  end
for i ← 1 to N_1 do
  /* Individual selection */
  for j ← 1 to N_2 do
    J_N(i, j) ~ Mult(\{\tilde{ω}_N(i, j')\}_{j'=1}^{N_1});
  end
  /* Mutation */
  for j ← 1 to N_2 do
    ξ_N^{(p+1)}(i, j) ~ R_p(ξ_N^{(p)}(I_N(i), J_N(i, j)), ·);
    ω_N^{(p+1)}(i, j) ← w_p(ξ_N^{(p)}(I_N(i), J_N(i, j)), ξ_N^{(p+1)}(i, j));
  end
  Ω_N^{(p+1)}(i) ← ∑_{j=1}^{N_2} Ω_N^{(p+1)}(i, j)/N_2;
end

Theorem 1 ([3]). For all n ∈ ℤ and h ∈ F_b(ℤ),
η_n h → h as N → ∞.

Impose the following additional assumption.

(S) For all β > 0, N_1 exp(−βN_2) → 0 as N → ∞.
Under (S), which guarantees that N_1 grows only subexponentially fast with respect to N_2, one may derive the following CLT.

Theorem 2 ([3]). Assume (S). Then for all n ∈ ℤ,
the random variable \{CV^2(\{Ω_N^{(h)}(i)\}_{i=1}^{N_1}) > τ\} has a deterministic limit ε_n in probability. Moreover, for all h ∈ F_b(ℤ), as N → ∞,
\[ \sqrt{N}(η_n h − η_0 h) \rightarrow N(0, V_n(h) + V_0^2(h)), \]
where V_0(h) = η_0(h − η_0 h)^2, V_0 = 0, and
\[ V_n(h) = ∑_{p=0}^{n-1} ∑_{l=0}^{n-1} ε_p R_l (w_l^2 Q_{l+1} Q_{l+2} · · · Q_{n-1} (h − η_0 h)^2). \]

Note that the B^2 algorithm, which selects systematically the islands, is a particular case of the B^2ASIL algorithm for which τ = 0 (and hence ε_n = 1) for all n ∈ ℤ. We may hence deduce the asymptotic variance \( \sigma^2_0(h) \) of the B^2 algorithm:
\[ \sum_{l=0}^{n-1} (n-l)^{n-l} R_l (w_l^2 Q_{l+1} Q_{l+2} · · · Q_{n-1} (h − η_0 h)^2). \]

In this part we discuss conditions under which interaction on the island level is desirable. The case of independently evolving islands is a particular case of the B^2ASIL algorithm with τ = ∞ (and hence ε_n = 0 for all n ∈ ℤ). We hence deduce, from Theorem 2, that the asymptotic variance of the estimator non-interactive bootstrap filters is, when N is large, V_n/N, i.e., inversely proportional to N.
In addition, it was shown in [1] that the asymptotic bias of this estimator is B_n/N_2, i.e., inversely proportional to N_2 and where B_n is some number depending on the model, which is presumably defined in [1]. Consequently, island interaction implies an additional, positive variance term while decreasing
the bias, and a trade-off between bias and variance has to be made to decide when island interaction is beneficial. For this purpose, we compare the mean squared errors (MSE) when the islands interact and when they are kept independent. The MSE for independent islands is given by \( \frac{V_n}{N} + \frac{B_2^2}{N^2} \), whereas the MSE of the B^2ASIL algorithm is given by \( \frac{(V_n + \hat{V}_n^2)}{N} \). Therefore,

\[
\frac{V_n + \hat{V}_n^2}{N} < \frac{V_n}{N} + \frac{B_2^2}{N^2} \iff N_2 < \frac{B_2^2}{V_n} N_1.
\]

Consequently, the B^2ASIL algorithm outperforms a bank of independently evolving bootstrap filters when the number of particles \( N_2 \) within each island is small compared to the number of islands \( N_1 \); the interaction reduces the bias (which is independent of \( N_1 \) when the islands are kept independent). On the contrary, when \( N_2 \) is larger than \( N_1 \), the variance increase introduced by the interaction on the island level may be larger than the bias reduction.

5. APPLICATION

We consider the stochastic volatility model

\[ X_{p+1} = \alpha X_p + \sigma U_{p+1}, \quad Y_p = \beta \exp(X_p/2) V_p, \]

where \( X_0 \sim \mathcal{N}(0, \sigma^2/(1 - \alpha^2)) \), \( \{U_p\}_{p \in \mathbb{N}} \) and \( \{V_p\}_{p \in \mathbb{N}} \) are independent sequences of mutually independent standard Gaussian random variables (independent of \( X_0 \)). We based our simulations on \( n = 100 \) observations generated under the model dynamics \((\alpha, \sigma, \beta) = (0.98, 0.5, 1)\). We estimate the means of the predictive distributions \( X_p \mid \{Y_0 = y_0, \ldots, Y_{p-1} = y_{p-1}\} \), for \( p \in \{0, 100\} \), a problem that can be cast into the Feynman-Kac framework by setting, for all \( p \in \mathbb{N}, \nabla

\[
Q_p(x_p, \Delta x_{p+1}) = \frac{1}{256} \exp \left( \frac{\gamma^2}{\sigma^2} \exp(-x_p) \right) - \frac{x_p}{2} - \frac{(x_{p+1} - x_p)^2}{2\sigma^2} \right) \Delta x_{p+1}.
\]

As a reference, we computed such predictive expectations using a single run of the bootstrap filter with \( 10^6 \) particles. Figure 1 displays box plots based on 250 replicates of the different algorithms for different values of \( N_1 \) and \( N_2 \). Table 1 reports the average numbers of island interactions for B^2ASIL over the 250 simulations. These numbers should be compared to \( n = 100 \) for B^2, since island interaction is systematic in this case. For a given number of islands, the number of island interactions decreases for B^2ASIL when the island size grows. Moreover, as soon as the number of particles \( N_2 \) of each island is large enough, B^2ASIL is no longer resampling the islands. Figure 2 displays CPU times for a \( N_2 = 100 \) and different numbers of islands. CPU time for the B^2ASIL is bigger than for independent bootstraps (due to the additional resampling step), and increases proportionally to the island number. Also notice that the parallel architecture of B^2ASIL reduces significantly CPU times compared to a classic bootstrap with \( N_1 N_2 \) particles.

6. STABILITY OF THE B^2 ALGORITHM

When studying the numerical stability of the B^2 algorithm we will work under the following strong

![Fig. 1. Comparison of different interactions across the islands for the Stochastic volatility model (1) Independent bootstraps; (2) B^2ASIL; (3) B^2.]

<table>
<thead>
<tr>
<th>( N_1 )</th>
<th>( N_2 )</th>
<th>10</th>
<th>100</th>
<th>1000</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>30.1</td>
<td>35.14</td>
<td>36.108</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>10.9</td>
<td>12.29</td>
<td>12.096</td>
</tr>
<tr>
<td>100</td>
<td>100</td>
<td>1.5</td>
<td>1.86</td>
<td>1.956</td>
</tr>
<tr>
<td>1000</td>
<td>1000</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 1. Number of island interactions for the B^2ASIL algorithm.
**Corollary 3.** Suppose (M). Then for all $n \in \mathbb{N}$ and $h \in F_b(\mathcal{X})$,

$$\sigma_n^2(h) \leq w_+ \frac{\text{osc}_n^2(h)}{(1-\rho)^2(1-\rho^2)^2}.$$

**References**


