Adaptive Multiple Importance Sampling

JEAN-MARIE CORNUET

Centre de Biologie et Gestion des Populations, INRA

JEAN-MICHEL MARIN

Institut de Mathématiques et Modélisation de Montpellier (UMR CNRS 5149), Université Montpellier 2

ANTONIETTA MIRA

Department of Economics, University of Lugano

CHRISTIAN P. ROBERT

Université Paris Dauphine, CEREMADE, IUF and CREST

ABSTRACT. The Adaptive Multiple Importance Sampling algorithm is aimed at an optimal recycling of past simulations in an iterated importance sampling (IS) scheme. The difference with earlier adaptive IS implementations like Population Monte Carlo is that the importance weights of all simulated values, past as well as present, are recomputed at each iteration, following the technique of the deterministic multiple mixture estimator of Owen & Zhou (*J. Amer. Statist. Assoc.*, 95, 2000, 135). Although the convergence properties of the algorithm cannot be investigated, we demonstrate through a challenging banana shape target distribution and a population genetics example that the improvement brought by this technique is substantial.

Key words: adaptive importance sampling, banana shape target, deterministic mixture weights, particle filters, population genetics, population Monte Carlo, sequential Monte Carlo

1. Introduction

Importance sampling (IS) [see for instance Ripley (1987)] is a well-established method used to overcome the difficulties connected with the complexity of simulating from a target distribution Π. Its shortcomings are also well-documented, first and foremost the degradation of its performances against the dimensionality of the problem. Given an importance distribution Q, such that Π is absolutely continuous with respect to Q, IS is based on samples $\mathbf{y}_i \sim Q$. The corresponding importance weights $\omega_i = \pi(\mathbf{y}_i)/q(\mathbf{y}_i)$ are defined in terms of $\pi(\cdot)$ and $q(\cdot)$, the densities of, respectively, the target and the importance distributions with respect to the same dominating measure v. The distribution of those weights customarily deteriorates as the dimension of $\mathbf{y}_i$ increases ($\mathbf{y}_i$ takes values in $\mathbb{R}^p$). Since, in practical settings, the fine tuning of the importance distribution against the target is difficult, alternative Markov chain Monte Carlo approaches have often been advocated as being more appropriate for large dimensional problems (see Robert & Casella, 2004) but recent attempts have been made to construct importance functions that automatically adapt to the target distribution based on earlier importance samples (see, e.g. Ortiz & Kaelbling, 2000; Liu *et al.*, 2001; Pennanen & Koivu, 2004; Rubinstein & Kroese, 2004). Those methods are called adaptive importance sampling (AIS), but they also relate to particle filters (Gordon *et al.*, 1993; Doucet *et al.*, 2001) and sequential Monte Carlo methods (Doucet *et al.*, 2000; Chopin, 2002; Del Moral *et al.*, 2006).

There are many different strategies or devizing AIS algorithms. For instance, the generic Population Monte Carlo (PMC) scheme of Cappé *et al.* (2004) can be implemented as the *D*-kernel (Douc *et al.*, 2007a,b) algorithm, whose goal is to fit a mixture of *D* given kernels

to the target in terms of either minimum variance or minimum Kullback–Leibler divergence. Although this algorithm is shown to converge to the optimal solution (meaning either minimum variance or minimum Kullback–Leibler divergence) within the class of D-kernels, it is restrictive to a specific type of importance distributions that may fail to properly represent the target.

In this paper, we propose a novel perspective to pool together importance samples from different IS distributions. Those various importance samples $\mathbf{y}_i^t \sim Q_t$ ($0 \leq t \leq T, 1 \leq i \leq N_t$) are associated with importance weights

$$\omega_i^t = \pi(\mathbf{y}_i^t)/q_t(\mathbf{y}_i^t), \tag{1}$$

where q_t and π are proper densities. Although those T samples can be crudely merged by keeping these original importance weights (Robert & Casella, 2004, chapter 14), there exists a more refined and stabilizing alternative called *deterministic multiple mixture* due to Veach & Guibas (1995) and popularized by Owen & Zhou (2000).

This alternative solution is similar to the defensive sampling approach of Hesterberg (1995) in that it modifies the denominator of the importance weight ω_i^t from the density value in $\mathbf{y}_i^t$, $q_t(\mathbf{y}_i^t)$, to a mixture of all the densities that produced the T different samples, namely

$$\frac{1}{\sum_{j=0}^{T} N_j} \sum_{l=0}^{T} N_l q_l(\mathbf{y}_i^t), \tag{2}$$

resulting in the (so-called deterministic) mixture weight

$$\omega_i^t = \pi(\mathbf{y}_i^t) \left/ \frac{1}{\sum_{j=0}^{T} N_j} \sum_{l=0}^{T} N_l q_l(\mathbf{y}_i^t) \right. . \tag{3}$$

This idea has originally been proposed by Veach & Guibas (1995) and is validating by the unbiasedness property

$$\mathbb{E}\left[\frac{1}{\sum_{j=0}^{T} N_j} \sum_{t=0}^{T} \sum_{i=1}^{N_t} \omega_i^t h(\mathbf{y}_i^t) \right] = \sum_{t=0}^{T} N_t \int h(\mathbf{y}) \frac{\pi(\mathbf{y})}{\sum_{l=0}^{T} N_l q_l(\mathbf{y})} q_t(\mathbf{y}) v(\mathrm{d}\mathbf{y})$$

$$= \int h(\mathbf{y}) \pi(\mathbf{y}) v(\mathrm{d}\mathbf{y}) = \mathbb{E}_\Pi[h(\mathbf{y})]. \tag{4}$$

The name *deterministic mixture weights* stems from the fact that the weights of the mixture (2) are neither estimated nor varying over time (which is coherent given that the algorithm is not sequential). This is a major difference with the PMC schemes of Douc *et al.* (2007a,b) where the weights of the proposals are optimized against an efficiency criterion like the Kullback–Leibler divergence. Deterministic mixture thus misses this adaptive feature, and our proposal called Adaptive Multiple Importance Sampling (AMIS) aims at bridging this gap.

When compared with the previous works on multiple mixtures, the novelty in AMIS is that the family (Q_t) of IS distributions is constructed sequentially and adaptively. This means that the IS distribution used at each iteration t ($1 \leq t \leq T$) is derived from the past $t-1$ importance weighted samples. More precisely, at each step t:

(i) the importance weights of all (present and past) simulated variables $\mathbf{y}_i^l$ ($1 \leq l \leq t$, $1 \leq i \leq N_t$) are modified, based on the current collection of proposals (IS distributions) $(Q_l)_{0 \leq l \leq t}$, and

(ii) the entire collection of importance samples partakes to the construction of the next importance function, $Q_{t+1}+$.

Note that, while (ii) is a classical feature of PMC algorithms, most implementations that derive Q_t from past iterations (Douc *et al.*, 2007b; Cappé *et al.*, 2008) restricted to use only samples produced at the previous generation, $t-1$. However, using the entire past of the simulation process provides a natural stabilization that speeds up convergence, but require a much more involved mathematical machinery. A similar type of methodology has been independently studied by Raftery & Bao (2010).

In most practical settings where IS is implemented, primarily in Bayesian estimation, a self-normalized estimator is used instead, because the density π of the target is known only up to a normalizing constant. In such cases, the importance weights can be evaluated only up to this normalizing constant and thus need to reweighted by the sum of the weights. By construction, the self-normalized estimator does not depend on this constant. In the forecoming examples, we therefore always use the self-normalized AMIS estimator, even for the benchmark banana shape target considered in section 6.

The plan of the paper is as follow: we detail the reasons for promoting multiple mixture IS in section 2 and analyse some associated algorithms in section 3, while discussing their theoretical properties in section 5. The performances of the AMIS algorithm are tested in section 6 over a challenging banana shape target distribution and in section 7 over a realistic population genetic application. We stress that the latter has motivated the development of the proposed methodology. Indeed, the likelihood of a genetic model most often is not tractable and regardless of the approximation method used, its derivation involves a non-negligible cost. We point out that Sirén *et al.* (2010) have resorted to our AMIS algorithm to handle complex population genetics models, avoiding the dramatic consequences of a poor first proposal.

2. Multiple mixtures

The modification in the importance weights from the original ratios (1) to the mixture ratios (3) may sound surprising or even paradoxical in that the simulated values (and therefore the distributions used to simulate those) have not changed. We thus detail in this section the motivations for using multiple mixtures. There exists a fundamental methodological difficulty in using several importance functions at once. Indeed, if Π is the target density and $Q_0, \ldots, Q_T$ are T different importance functions, samples $\mathbf{y}_1^0, \ldots, \mathbf{y}_{N_0}^0, \ldots, \mathbf{y}_1^T, \ldots, \mathbf{y}_{N_T}^T$ that are simulated from these importance functions, with associated standard importance weights $\omega_i^t = \pi(\mathbf{y}_i^t)/q_t(\mathbf{y}_i^t)$, can be merged together in that the empirical distribution function

$$\sum_{t,i} \omega_i^t \delta_{\mathbf{y}_i^t}(\mathbf{y}) \Big/ \sum_{t,i} \omega_i^t$$

produces in the marginal sense an output approximatively distributed from the target π. Unfortunately, this property is not sufficient to ensure that the resulting sample performs satisfactorily. For instance, if one of the importance functions q_t is associated with an infinite variance in the weights ω_i^t, i.e. if $\mathbb{E}[(w_i^t)^2] = +\infty$ for one $0 \leq t \leq T$, the potentially very large weights resulting from this importance experiment will remain very large in the cumulated sample, no matter how efficient the other importance functions are. Therefore, the poorly performing sample will overwhelmingly dominate the other samples in the final approximation and thus ruin the overall performances of the method. The conclusion of this point is that the raw mixing of importance samples and of their importance weights, when using different proposals, can be quite harmful, when compared with using a single sample, even when most proposals are efficient.

As discussed at large in Owen & Zhou (2000), using a deterministic mixture as a representation of the production of the simulated sample has the potential to exploit the most efficient proposals in the sequence $Q_0, \ldots, Q_T$ without rejecting any simulated value nor sample, while reducing the variance of the corresponding estimators. The poorly performing importance functions are simply eliminated through the erosion of their weights

$$\pi(\mathbf{y}_i^t) \bigg/ \frac{1}{\sum_{j=0}^{T} N_j} \sum_{l=0}^{T} N_l q_l(\mathbf{y}_i^t)$$

as T increases. Indeed, for all $N_i \geq 1$ not necessarily equals, if q_0 is the poorly performing proposal, while the q_ls $(l > 1)$ are good approximations of π, for a value $\mathbf{y}_i^0$ such that $\pi(\mathbf{y}_i^0)/q_0(\mathbf{y}_i^0)$ is large, because $q_0(\mathbf{y}_i^0)$ is small, $\pi(\mathbf{y}_i^0)/\{N_0 q_0(\mathbf{y}_i^0) + \cdots + N_T q_T(\mathbf{y}_i^0)\}$ will behave like $\pi(\mathbf{y}_i^0)/\{N_1 q_1(\mathbf{y}_i^0) + \cdots + N_T q_T(\mathbf{y}_i^0)\}$ and decrease to zero as T increases.

3. The Adaptive Multiple Importance Sampling algorithm

As explained in the introduction, the idea at the core of the AMIS algorithm is that for each time-step t, we should update not only the weights ω_i^t of the N_t current particles, $\mathbf{y}_i^t$, but also the weights ω_i^l of all past particles $\mathbf{y}_i^l, 0 \leq l \leq t - 1$. Our algorithm can thus be interpreted as a Rao–Blackwell type of IS where the whole sample of $\sum_{j=0}^{T} N_j$ points can be envisioned of as being homogeneously sampled from a deterministic mixture made of the overall sum of proposals. (Once again, the term *deterministic mixture* is a misnomer in that the overall sample is not the outcome of a mixture simulation.)

The major difference with various PMC versions (Cappé *et al.*, 2004; Douc *et al.*, 2007a,b; Cappé *et al.*, 2008) is that every single simulated value is recycled and reweighted at every step of our iterative algorithm by virtue of selecting the appropriate deterministic mixture. Indeed, at each iteration t of the algorithm, a new AIS distribution is constructed by using not only the particles corresponding to the current iteration, but all the weighted particles, based on a well-chosen efficiency criterion as in earlier PMC versions (Cappé *et al.*, 2008). In the most standard case when the proposal Q_t is parameterized, i.e. when Q_t is of the form $Q(\theta_t)$ within a parametric family of distributions $\{Q(\theta), \theta \in \Theta\}$, the adaptivity consists in estimating θ_t by $\hat{\theta}_t$ at each iteration, using all the weighted samples accumulated so far; this estimation is obtained by using specific criterion like moment matching, variance minimization or Kullback–Leibler minimization.

A pseudo-code representation of the generic AMIS algorithm is given as follows:

Algorithm 1. Generic AMIS

At iteration $t = 0$,

(1) Independently generate N_0 particles $\mathbf{y}_i^0$ $(1 \leq i \leq N_0)$ from Q_0.
(2) For $1 \leq i \leq N_0$, compute

$$\delta_i^0 = N_0 q_0(\mathbf{y}_i^0) \quad \text{and} \quad \omega_i^0 = \pi(\mathbf{y}_i^0)/q_0(\mathbf{y}_i^0).$$

(3) Compute the importance sampling parameter estimate $\hat{\theta}^0$ of the parametric family $\{Q(\theta), \theta \in \Theta\}$ using the weighted particles

$$\left(\{\mathbf{y}_1^0, \omega_1^0\}, \ldots, \{\mathbf{y}_{N_0}^0, \omega_{N_0}^0\} \right)$$

and a well-chosen estimation criterion.

At iteration $t = 1, \ldots, T$

(1) Independently generate N_t particles $\mathbf{y}_i^t$ $(1 \leq i \leq N_t)$ as $x_i^t \sim Q(\hat{\boldsymbol{\theta}}^{t-1})$.

(2) For $1 \leq i \leq N_t$, compute the multiple mixture at x_t^i

$$\delta_i^t = N_0 q_0(\mathbf{y}_i^t) + \sum_{l=1}^{t} N_l q(\mathbf{y}_i^t; \hat{\boldsymbol{\theta}}^{l-1})$$

and derive the importance weight of particle $\mathbf{y}_i^t$,

$$\omega_i^t = \pi\left(\mathbf{y}_i^t\right) \bigg/ \left[\delta_i^t \bigg/ \sum_{j=0}^{t} N_j \right].$$

(3) For $0 \leq l \leq t-1$ and $1 \leq i \leq N_l$, update the past importance weights as

$$\delta_i^l \longleftarrow \delta_i^l + N_l q(\mathbf{y}_i^l; \hat{\boldsymbol{\theta}}^{t-1}) \qquad \text{and} \qquad \omega_i^l \longleftarrow \pi\left(\mathbf{y}_i^l\right) \bigg/ \left[\delta_i^l \bigg/ \sum_{j=0}^{t} N_j \right].$$

(4) Compute the parameter estimate $\hat{\boldsymbol{\theta}}^t$ using all the weighted particles

$$\left(\{\mathbf{y}_1^0, \omega_1^0\}, \ldots, \{\mathbf{y}_{N_0}^0, \omega_{N_0}^0\}, \ldots, \{\mathbf{y}_1^t, \omega_1^t\}, \ldots, \{\mathbf{y}_{N_t}^t, \omega_{N_t}^t\} \right)$$

and the same estimation criterion.

After T iterations of the AMIS algorithm, for any Π-integrable function h, the self-normalized AMIS estimator of $\mathbb{E}_\Pi(h(\mathbf{y})) = \int h(\mathbf{y})\pi(\mathbf{y})\nu(\mathrm{d}x)$ is as follows:

$$\widehat{\mathbb{E}_\Pi(h(\mathbf{y}))} = \frac{1}{\sum_{t=0}^{T} \sum_{i=1}^{N_t} \omega_i^t} \sum_{t=0}^{T} \sum_{i=1}^{N_t} \omega_i^t h(\mathbf{y}_i^t). \tag{5}$$

As the above algorithm is set in generic terms, we describe a first special case that applies to many settings and can be seen as a vanilla AMIS algorithm. As in the most recent PMC algorithm of Cappé *et al.* (2008), the proposal distribution Q is a Student's t proposal, $\mathcal{T}_3(\mu, \Sigma)$ whose mean μ and covariance Σ parameters are updated by estimating both first moments of the target distribution Π using self-normalized AMIS estimators: $\hat{\boldsymbol{\theta}}^t = (\mu^t, \Sigma^t)$ and

$$\hat{\mu}^t = \frac{\sum_{l=0}^{t} \sum_{i=1}^{N_l} \omega_i^l \mathbf{y}_i^l}{\sum_{l=0}^{t} \sum_{i=1}^{N_l} \omega_i^l} \quad \text{and} \quad \hat{\Sigma}^t = \frac{\sum_{l=0}^{t} \sum_{i=1}^{N_l} \omega_i^l (\mathbf{y}_i^l - \hat{\mu}^t)(\mathbf{y}_i^l - \hat{\mu}^t)^{\mathrm{T}}}{\sum_{l=0}^{t} \sum_{i=1}^{N_l} \omega_i^l}. \tag{6}$$

Note that the degrees of freedom of the t distribution are always set to 3 as the lowest value allowing for finite first moments but they could also be estimated at each iteration. Moreover, instead of using the previous 'moments matching' criterion, we can also used the Kullback–Leibler divergence between Π and Q to choose the parameter $\theta = (\mu, \Sigma)$,

$$\mathrm{div}(\Pi, Q(\theta)) = \int \log \frac{\pi(\mathbf{y})}{q(x; \theta)} \pi(\mathbf{y})\nu(\mathrm{d}\mathbf{y}).$$

Here, the best choice for the parameter θ is the maximum likelihood estimate of θ where the observations are weighted by their corresponding importance weight. These two different strategies give essentially the same results.

Quite obviously and as illustrated by the next section, more elaborate proposals are possible, depending on the information available on Π. For instance, if the potential for multimodality of the target Π is high enough, a mixture of Student's t distributions as in Cappé *et al.* (2008) would be more appropriate. When dealing with a Bayesian hierarchical model, creating classes (or blocks) of components of the parameter in agreement with the

hierarchical levels (as in Gibbs sampling) and designing the Student's t proposals block by block should also be more efficient.

Similarly, matching the expectation and the covariance structure of the Student's proposal distribution with both first moments of the target distribution is only one among many efficiency criteria that can be used to calibrate the parameters of the proposal distribution at each step of the algorithm. For instance, as done in the next section, we can alternatively minimize the Kullback–Leibler divergence between the target and the proposal distribution following the approach of Cappé *et al.* (2008).

Although we do not elaborate on this possible improvement, note also that, once the weighted sample based on $\sum_{t=0}^{T} N_t$ simulations is obtained, it is possible to apply a final clustering (standard) algorithm on this sample, based on a Gaussian mixture representation. Those clusters can be used to estimate local covariance and mean structures and then simulate a final and global sample based on the cluster representation but using Student's t distributions. Because all weights are controlled, we can then merge this final sample with the sequence of earlier samples without losing the deterministic representation.

A special version of interest of the AMIS algorithm is based on the used of mixtures of multivariate Gaussian densities. That is

$$q(\mathbf{y}; \boldsymbol{\theta}) = \sum_{i=1}^{k} \rho_i \varphi(\mathbf{y}; \mu_i, \Sigma_i), \qquad \sum_{i=1}^{k} \rho_i = 1,$$

where $\varphi(\cdot; \mu, \Sigma)$ denotes a multivariate Gaussian density with mean μ and covariance matrix Σ, as in the D-kernel approach to PMC algorithms of Cappé *et al.* (2008). We also use the Kullback–Leibler divergence between Π and Q to choose the parameter $\boldsymbol{\theta} = (\rho_1, \ldots, \rho_k, \mu_1, \ldots, \mu_k, \Sigma_1, \ldots, \Sigma_k)$,

$$\mathrm{div}(\Pi, Q(\boldsymbol{\theta})) = \int \log \frac{\pi(\mathbf{y})}{q(\mathbf{y}; \boldsymbol{\theta})} \pi(\mathbf{y}) v(\mathrm{d}\mathbf{y}).$$

As already mentioned, the best choice for the parameter $\boldsymbol{\theta}$ is then the maximum likelihood estimate of $\boldsymbol{\theta}$. In the AMIS setting, the observations are weighted by their corresponding importance weight: at iteration t the whole sequence of samples $\mathbf{y}_i^l$ ($0 \leq l \leq t$) with their updated weights ω_i^l is used inside a weighted Expectation Maximization (EM) algorithm, which is solved using the mixmod software (Biernacki *et al.*, 2006). The number k of components used for the mixture can be either set in advance or, more realistically, estimated at iteration $t = 0$ by the Integrated Completed Likelihood (ICL) criterion of Biernacki *et al.* (2000) and a substantial number N_0 of iterations. We do not reproduce the earlier pseudo-codes for this special case as the differences are minimal. Note that the extension to a mixture of t densities is equally feasible since there exists a corresponding EM algorithm (Peel & McLachlan, 2000).

4. Initialization

A primary difficulty with adaptive importance algorithms is that the starting distribution has a major impact on the resulting performances of those algorithms. Due to the 'what-you-get-is-what-you-see' nature of such algorithms, it is quite difficult to recover from a poor starting sample, the adaptivity focussing only on the visited parts of the simulation space. Therefore, we strongly require that a significant part of the computing effort be spent on the initialization stage.

In order to calibrate this computing effort, we use the effective sample size (ESS). For a sample of size N_0 based on the importance distribution Q_0, the ESS is defined by $N_0 / \{1 + \mathbb{V}_{Q_0}[\pi(\mathbf{y})/q_0(\mathbf{y})]\}$ (Hesterberg, 1995; Liu, 2001), and it corresponds to the size of an

equivalent i.i.d. sample simulated from Π. This measure of efficiency does not depend on h and, in practice,

$$\mathbb{V}_{Q_0}[\pi(\mathbf{y})/q_0(\mathbf{y})] = \int \left\{ \pi(\mathbf{y})/q_0(\mathbf{y}) - \mathbb{E}_{Q_0}[\pi(\mathbf{y})/q_0(\mathbf{y})] \right\}^2 q_0(\mathbf{y})\nu(\mathrm{d}\mathbf{y})$$

can be estimated using the coefficient of variation of the importance weights.

The initialization solution we propose proceeds through two steps:

  (i) Independently generate N_0 uniform samples on the p-dimensional hyper-cube, $U_1, \cdots, U_{N_0}$.
 (ii) Use an inverse logistic transformation, with scale parameter s, to map these points on $\mathbb{R}^p$, independently for each coordinate.
(iii) Maximize the ESS of the points with respect to the scale parameter of the logistic distribution and obtain s^*.
 (iv) Use as starting cohort of particles $F^{-1}(U_i; s^*)$, $i = 1, \cdots, N_0$ where $F^{-1}(u; s) = s \log(u/(1-u))$ (vectorized expression $u \in \mathbb{R}^p$).

Note that to maximize the ESS is equivalent to minimize the variance of the importance weights. Moreover, to maximize the ESS, we only need simulate a single logistic sample since we can adapt the scale of this sample by mere multiplication. Obviously, this solution is far from fool-proof, and we favour an informed alternative implementation provided items of information on the target distribution are available. Those items may obviously be provided by multiple pilot runs.

Nelder & Mead's (1965) algorithm is used to maximize the ESS. This simplex method depends on the comparison of the ESS values at the $p+1$ vertices of a general simplex, followed by the replacement of the vertex with the smallest value by another point. The simplex keeps adapting to the local landscape and converges to the global maximum.

5. Convergence issues and tuning

Although establishing unbiasedness and convergence of the deterministic mixture estimator of Owen & Zhou (2000) is relatively straightforward, the introduction of an adaptive mechanism in the construction of the sequence of proposals highly complicates handling both properties. First, the estimator is no longer unbiased and its convergence (in T for a fixed values of N_t) cannot be established without imposing compactness restrictions on the simulation space or upper bounds on the target density.

In order to detail convergence difficulties, we concentrate on the Student's t version of the AMIS algorithm. Furthermore, we only consider the extreme case $N_t = 1$, meaning that each iteration of the algorithm only processes a single new simulated value: the proposal is then updated after each new iteration. We also simplify the update of the parameters of the Student's t proposal by restricting learning to the mean $\hat{\mu}^t$, the covariance matrix being set to an arbitrary value. This clearly is a formalized setting that we do not advocate in practice.

The density of the Student's t distribution with 3 degrees of freedom and mean mu is denoted $t_3(\mathbf{y}; \mu)$. The update of μ after iteration t is then

$$\hat{\mu}^t = u_{t+1}(\mathbf{y}_{0:t}) = \sum_{k=0}^{t} \frac{\pi(\mathbf{y}_k)\mathbf{y}_k}{q_0(\mathbf{y}_k) + \sum_{i=1}^{t} t_3(\mathbf{y}_k; u_i(\mathbf{y}_{0:i-1}))},$$

where $u_1(\mathbf{y}_0) = \pi(\mathbf{y}_0)\mathbf{y}_0/q_0(\mathbf{y}_0) = \hat{\mu}^0$.

First, the unbiasedness of the estimator $\hat{\mu}^t$ for every $t > 1$ does not follow from the arguments found in the original version of Owen & Zhou (2000) because of the dependence of the importance weight of $\mathbf{y}_t$ on subsequent $\mathbf{y}_j$s ($j > t$). Indeed, for $t \geq 1$, we have

$$\mathbb{E}[\hat{\mu}^t] = \sum_{k=0}^{t} \mathbb{E}\left[\frac{\pi(\mathbf{y}_k)\mathbf{y}_k}{q_0(\mathbf{y}_k) + \sum_{i=1}^{t} t_3(\mathbf{y}_k; u_i(\mathbf{y}_{0:i-1}))}\right]$$

$$= \sum_{k=0}^{t} \int \frac{\pi(\mathbf{y}_k)\mathbf{y}_k}{q_0(\mathbf{y}_k) + \sum_{i=1}^{t} t_3(\mathbf{y}_k; u_i(\mathbf{y}_{0:i-1}))} t_3(\mathbf{y}_k; u_k(\mathbf{y}_{0:k-1}))\, \mathrm{d}\mathbf{y}_k$$

$$\times g_k(\mathbf{y}_{0:k-1})\, \mathrm{d}\mathbf{y}_{0:k-1} h_k(\mathbf{y}_{k+1:t}\,|\,\mathbf{y}_{0:k})\, \mathrm{d}\mathbf{y}_{k+1:t},$$

where $g_k(\mathbf{y}_{0:k-1})$ is the joint distribution of the past simulations and $h_k(\mathbf{y}_{k+1:t}\,|\,\mathbf{y}_{0:k})$ is the conditional distribution of the future simulations given the current and past ones. Due to this latter term, the full conditional distribution of $\mathbf{y}_k$ given the past and future simulations $\mathbf{y}_{0:k-1}$ and $\mathbf{y}_{k+1:t}$ is no longer $t_3(\mathbf{y}_k; u_k(\mathbf{y}_{0:k-1}))$ and this modification implies that $\hat{\mu}^t$ is biased. Furthermore, the dependence of this bias on t is so intricate that we cannot manage the asymptotic bias. A similar impossibility occurs when studying the variance, hence preventing a theoretical conclusion about the convergence properties of the AMIS algorithm. Moreover, the standard convergence results on triangular arrays (Douc & Moulines, 2008) do not apply here, contrariwise to the PMC algorithm (Douc *et al.*, 2007a). Note that a simple if artificial modification of the AMIS algorithm brings a straightwforward solution to the bias difficulty: when using an additional simulation thread for the calibration of the proposal distributions, the arguments of Owen & Zhou (2000) apply by first conditioning upon this second series.

Using exactly the same results as in Douc *et al.* (2007a), notably theorem A.1 on the convergence of triangular arrays, under very weak conditions, we obtain the following lemma for the AMIS algorithm:

Lemma 1. *When T and $N_0, \ldots, N_{T-1}$ are fixed, the estimator $\widehat{\mathbb{E}_\Pi(h(\mathbf{y}))}$ is converging in probability to $\mathbb{E}_\Pi(h(\mathbf{y}))$ when N_T goes to infinity.*

Indeed, the fact that all sample sizes N_t but the last one N_T are set to a given value means that the weights of the terms $\mathbf{y}_i^t$ ($0 \le t \le T-1$) converge to 0, while the bias in the weights of the $\mathbf{y}_i^T$ asymptotically vanishes, conditionally on the past samples. The weak conditions are related to the tail behaviour of the importance densities with respect to the target.

However, this setting is not the one in which AMIS should be used. The number of iterations T and the numbers of simulations N_t ($t = 1, \ldots, T$) should be related to the dimension p of the target distribution. We recommend to use N_t in the range 25–500: from 25 when p is small (typically $d = 1$ or $d = 2$) to 500 when p is large (typically $p = 20$). We tested this strategy for the AMIS algorithm on various target distributions. Note we used the default calibration $N_1 = N_2 = \cdots = N_T$. However, it should be more efficient to increase the numbers of simulations with the accuracy of the proposal distributions, $N_1 < N_2 < \cdots < N_T$, provided an automated scheme on the choice of the N_t can be found.

For instance, in the area of population genetics, Sirén *et al.* (2010) proposed an original Bayesian method for inferring population histories from unlinked single-nucleotide polymorphism. It is used on an approximation to the neutral Wright–Fisher diffusion that models random fluctuations in allele frequencies. Inference about the tree topology imply that the posterior distribution be marginalized over a drift parameter, which, for K populations, is a positive vector of dimension $2K - 2$. Sirén *et al.* (2010) circumvented this difficulty by resorting to the AMIS algorithm. They used a product of independent Beta distributions as the initial importance distribution (Q_0), then the following importance distributions ($q_t, t = 1, \ldots, T$) were defined as multivariate Student's distributions whose parameters are adapted at fixed interval. In their tests, they chose $N_0 = \sum_{t=0}^{T} N_t / 2$ and $T = 10$–200 depending on $\sum_{t=0}^{T} N_t$.

Typically, for simulated datasets where $K = 5$, they used $T = 200$ and $N_t = 50$ and, for an analysis of human data where $K = 7$, they computed the posterior probability of two topologies with $\sum_{t=0}^{T} N_t = 30{,}000$.

In connection with the dependence of the simulation numbers N_t on the dimension p of the target distribution, we note that the AMIS algorithm caters to highly different goals:

(i) To compute a numerical approximation of the expectation of a fixed function h, $I = h(x)\Pi(\mathrm{d}x)$.
(ii) To obtain an approximation of the marginal of a joint distribution.
(iii) To provide a global approximation of a sample from the target distribution Π.

Depending on the purpose for which the AMIS algorithm is used, the requested minimal value for the ESS will vary. For instance, if we want to approximate I for a specific function h, then the minimal value for the ESS depends on $\int (h(x) - \pi(h))^2 \Pi(\mathrm{d}x)$. If, instead, the goal is to approximate the target distribution Π, the minimal value for the ESS could be derived from the L_2 non-parametric estimation error, that is, $\int (\hat{\pi} - \pi)\mu(\mathrm{d}x)$ where $\hat{\pi}$ is a kernel density approximation of π. As a stopping rule, we propose to iterate the AMIS algorithm, i.e. to increase T, until the desired ESS is achieved.

The goal of the next section is to illustrate the fact that the AMIS algorithm can outperform a standard AIS solution on a benchmark target distributions. That is, with the same adaptive scheme, this algorith manages to get a significant improvement by pooling together all the simulated points in the sequential multiple mixture. Given that standard IS algorithms perform well only when T is small and when N_t is large, we also implemented the AMIS algorithm in this setup and chose $N_0 = 100{,}000$, $T = 10$ and $N_1 = \cdots = N_{10} = 10{,}000$.

In section 7, we instead consider a realistic population genetics example where the AMIS algorithm is implemented with $N_0 = 200{,}000$, $T = 2$ and $N_1 = N_2 = 200{,}000$. The initial IS distribution Q_0 is then the prior distribution. No optimization procedure related with the ESS is required in this case. Indeed, for such a target distribution, the region of relevance within the parameter space is easily reached, and we do not need many adaptation steps. However, the calculation of the target density is quite expensive and an involved recycling of the whole set of simulations is then relevant.

6. A banana shape target example

This evaluation of the performances of AMIS resorts to the benchmark target density of Haario *et al.* (1999, 2001), which can be calibrated as to become extremely challenging. The target density is based on a centred p-multivariate Gaussian, $\mathbf{y} \sim \mathcal{N}_p(0_p, \Sigma)$ with covariance matrix $\Sigma = \mathrm{diag}(\sigma^2, 1, \ldots, 1)$ which is twisted by a change of variable in the second coordinate from y_2 to $y_2 - b(y_1^2 - \sigma^2)$. Other coordinates remain unchanged. This change of variable leads to a twisted (or banana shaped) distribution that has expectation equal to 0 and uncorrelated components. Since the Jacobian of the twisting transformation is equal to 1, the target density is

$$\pi(\mathbf{y}) = f_{\mathcal{N}(0_p, \Sigma)}\left(y_1, y_2 + b(y_1^2 - \sigma^2), y_3, \ldots, y_p\right),$$

where $f_{\mathcal{N}(0_p, \Sigma)}(\cdot)$ denotes the density of the centred p-multivariate Gaussian distribution with covariance Σ. One of the appeals of this benchmark is to allow for various degrees of heavy tails through the choice of the parameter b.

In this example, we only consider a mild banana shape density, with $\sigma^2 = 100$ and $b = 0.03$. More twisted distributions, i.e. ones with fatter tails, can be obtained by calling for higher

values of b and/or σ^2. In this case, the target distribution satisfies $\mathbb{E}(y_i) = 0$ for all $i = 1, \ldots, p$, $\mathbb{V}(y_1) = 100$, $\mathbb{V}(y_2) = 19$ and $\mathbb{V}(y_i) = 1$ for all $i = 3, \ldots, p$.

For this target, we compare an iterative IS algorithm that uses the classical mixture version (as opposed to the deterministic mixture version) with the Gaussian mixture version of the AMIS algorithm. This reference algorithm, called AIS, thus also relies on past simulations for creating a new Gaussian mixture proposal, but it relies on usual importance weights. Given the recent work on PMC algorithms (Cappé *et al.*, 2008), this can be considered a state-of-the-art methodology for the comparison.

For both schemes, an initial sample of $N_0 = 10^5$ particles is simulated from a standard logistic distribution and rescaled component-wise to ensure a maximal ESS. In the following, $T = 10$ iterations and $N_t = 10,000$ particles ($1 \leq t \leq T$) are used. The clustering step fitting a mixture to the weighted samples is solved via the mixmod software (Biernacki *et al.*, 2000), with the number of components in the mixture being calibrated via the ICL criterion (Biernacki *et al.*, 2006) during the first iteration. It suggested resorting to a mixture of four components to correctly fit the banana shape target in two dimensions. Both schemes under comparison take approximatively the same computing time (depending of course on the dimension p of the problem) and produce 2×10^5 weighted particles. Note that, for $p = 20$, to maximize the ESS using the Nelder and Mead algorithm in the initialization step takes almost the same amount of time than $T = 10$ iterations of the AMIS algorithm with 10,000 particles per iteration.

The results of this experiment are reported in Table 1 and on Figs 1 and 2. These results all are consistent with a domination of the AMIS scheme. The gain in ESS is quite spectacular, but resulting from the strong stabilization brought by the AMIS averaging. The improvement in root mean square error shown in Table 1 typically varies with the target function as well as with the overall dimension p, but may go as far as a threefold reduction. The boxplots of the absolute errors convey the same message of a uniform domination by AMIS in this setting.

Table 1. *Mean square errors calculated over* 10 *replications of the AMIS and AIS schemes for different target functions for different values of p and in parenthesis the corresponding standard errors*

| Target function	p	AMIS	AIS
$\mathbb{E}(y_1) = 0$	5	0.00430 (0.00319)	0.00473 (0.00664)
	10	0.00408 (0.00469)	0.01221 (0.01224)
	20	0.00840 (0.00875)	0.03208 (0.03208)
$\mathbb{E}(y_2) = 0$	5	0.01044 (0.01486)	0.01342 (0.01275)
	10	0.04589 (0.04419)	0.05088 (0.03632)
	20	0.06409 (0.02552)	0.08461 (0.05381)
$\sum_{l=3}^{5} \mathbb{E}(y_i) = 0$	5	0.00002 (0.00003)	0.00009 (0.00008)
$\sum_{l=3}^{10} \mathbb{E}(y_i) = 0$	10	0.00009 (0.00014)	0.00044 (0.00074)
$\sum_{l=3}^{20} \mathbb{E}(y_i) = 0$	20	0.00028 (0.00053)	0.00177 (0.00343)
$\mathbb{V}(y_1) = 100$	5	6.795002 (6.72701)	15.41744 (14.34075)
	10	49.94052 (34.21143)	56.08176 (38.46109)
	20	67.24332 (47.74095)	94.42488 (58.44744)
$\mathbb{V}(y_2) = 19$	5	4.43871 (4.11778)	8.76941 (6.90886)
	10	14.18724 (6.54468)	25.85457 (12.67837)
	20	23.56200 (12.61588)	35.76413 (15.90980)
$\sum_{l=3}^{5} \mathbb{V}(y_i) = 3$	5	0.00004 (0.00003)	0.00014 (0.00019)
$\sum_{l=3}^{10} \mathbb{V}(y_i) = 8$	10	0.00019 (0.00034)	0.00069 (0.00104)
$\sum_{l=3}^{20} \mathbb{V}(y_i) = 18$	20	0.00212 (0.00245)	0.00413 (0.00613)
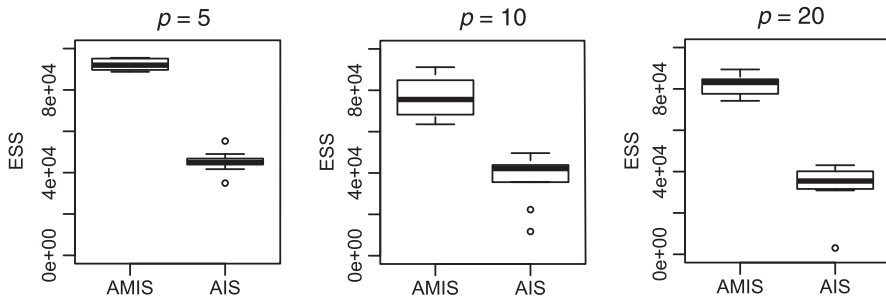
Fig. 1. Banana shape example: boxplots of the 10 replicate ESSs for the AMIS scheme (left) and the AIS scheme (right) for $p = 5, 10, 20$. The total number of particles is equal to 200,000.
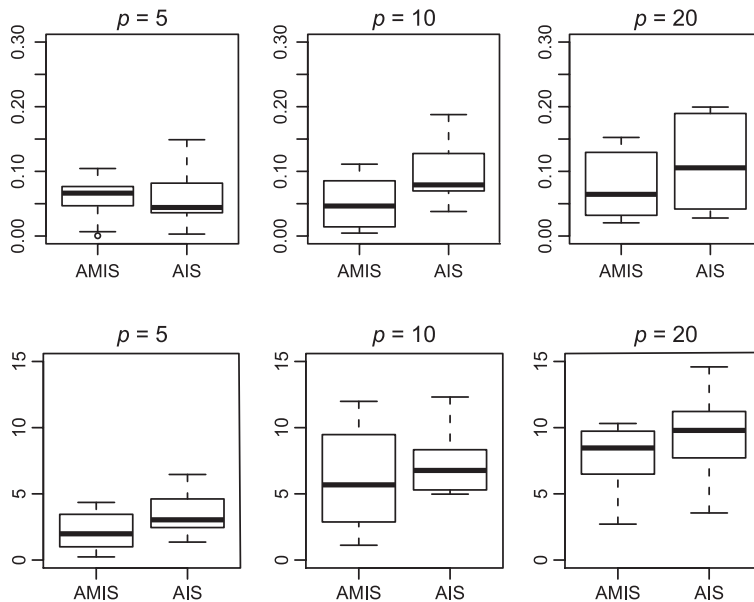

Fig. 2. Banana shape example: boxplots of the 10 replicate absolute errors associated to the estimations of $\mathbb{E}(y_1)$ (first line) and $\mathbb{V}(y_1)$ (second line) obtained by the AMIS and AIS schemes for $p = 5, 10, 20$.

7. An example from population genetics

Another illustration of the potential advantage in using the AMIS algorithm is now discussed. It addresses a realistic population genetics problem that essentially amounts to estimate parameters of an evolutionary scenario in which two populations have diverged from a common and unknown ancestral population. We suppose that we observed the genotypes at a single microsatellite locus of 50 diploid individuals sampled from each population. This locus is assumed to evolve according to the strict Stepwise Mutation model, i.e. when a mutation occurs, the number of repeats of the mutated gene increases or decreases by one unit with equal probability. After divergence, we also assume that populations do not exchange genes (no migration). The four parameters to estimate are the three effective population sizes (n_1, n_2, n_{Anc}) and the time of divergence (t_{div}), all scaled by the mutation rate (μ) of the locus : θ_1 ($= 4n_1\mu$), θ_2 ($= 4n_2\mu$), θ_A ($= 4n_{Anc}\mu$) and τ ($= t_{div}\mu$). The likelihood of this model is costly to obtain, which is why we selected this benchmark example. In a Bayesian framework, uniform
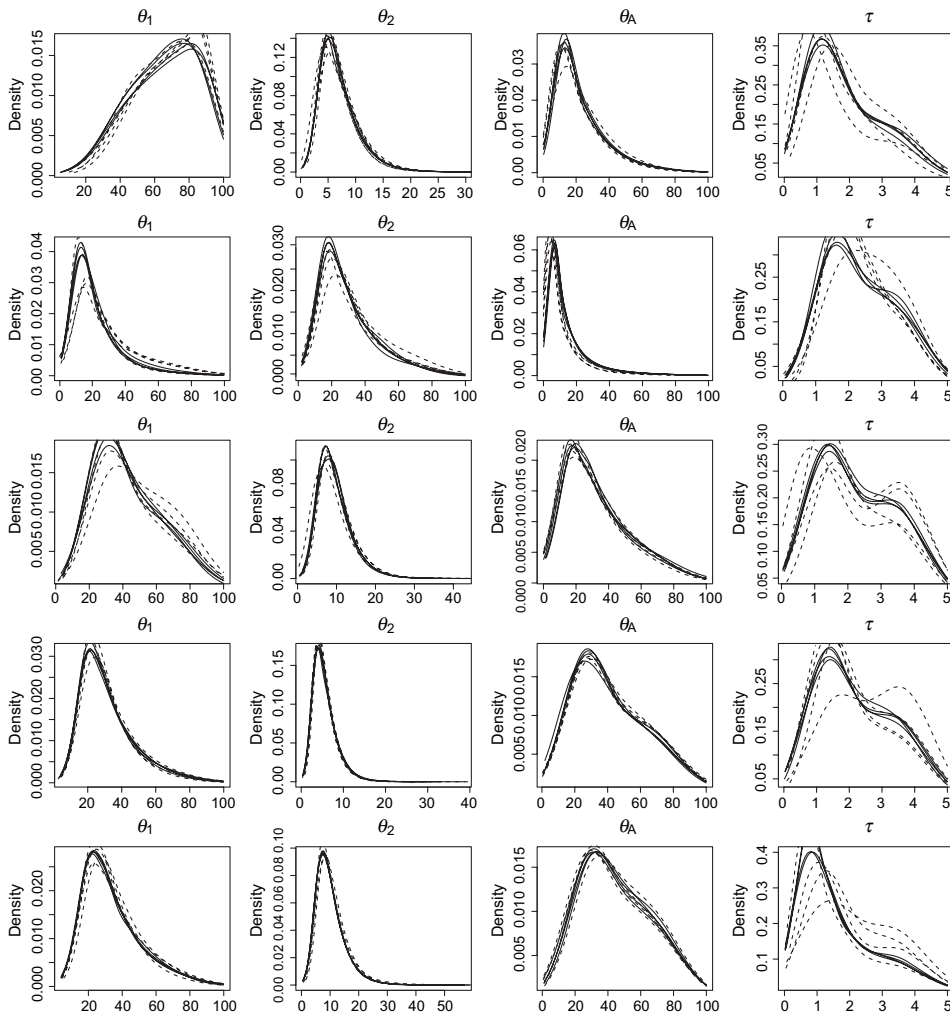
Fig. 3. Population genetics example: posterior distributions of the four parameters $(\theta_1, \theta_2, \theta_A, \tau)$ for five simulated datasets obtained through IS-AMIS (continuous line) and MCMC (dashed line). Each analysis has been repeated four times to evaluate the impact of repeatability.

priors $\mathcal{U}[0.1, 100]$ and $\mathcal{U}[0.005, 5]$ were chosen for the parameters θ and τ, respectively. Our target is the posterior distribution of $(\theta_1, \theta_2, \theta_A, \tau)$.

Five data files have been simulated with the software *DIYABC* (Cornuet *et al.*, 2008), with the following parameter values: $n_1 = n_{\text{Anc}} = 10{,}000$, $n_2 = 2000$, $t_{\text{div}} = 1000$ and $\mu = 0.0005$, leading to $\theta_1 = \theta_A = 20$, $\theta_2 = 4$ and $\tau = 0.5$. Each dataset has been processed twice.

The first analysis, used as a control, is based on a Monte Carlo Markov Chain (MCMC) run in which the gene tree of the sampled genes is updated together with the four demographic parameters. This has been performed with the software *IM* (Hey & Nielsen, 2004).

The second analysis combines the *AMIS* algorithm and an estimation of the likelihood based on IS for gene genealogies in the same way as Beaumont (2003) embedded an IS computation of the likelihood in a MCMC exploration of the parameter space. We note that the likelihood of a set of demographic parameters is computed by averaging importance weights of gene trees simulated event by event according to proposal distributions and

parameter values. Each gene tree is built in three steps looking backward in time: (i) between present time and time of divergence, lineages are coalesced or mutated following Stephens & Donnelly's (2000) algorithm, monitoring times of events as in Beaumont (2003), (ii) at time of divergence, remaining lineages of both populations are merged, and (iii) after divergence, the gene tree is completed according to the *SDPAC* algorithm of Cornuet & Beaumont (2007).

To assess the stability of the approximations provided by both methods, each analysis was repeated four times (i.e. with four different groups of random seeds for each dataset). Each MCMC (*IM*) was run as a single chain of 10^7 updates after a burn-in period of 10^6 updates. The IS-AMIS algorithm was run with $N_0 = 200,000$, $T = 2$, and $N_0 = N_1 = 200,000$. No optimization based on the ESS was required towards the calibration of the initial importance function: the prior distribution was deemed satisfactory. Indeed, the prior distribution is then sufficiently concentrated that there is no difficulty in finding the relevant region in the parameter space.

Both methods provided similar outputs as shown on Fig. 3, thus validating the IS-AMIS approach. However the major conclusion of this study is that, whereas each MCMC run lasted about 2 hours, the IS-AMIS execution lasted only approximately 20 minutes with a slightly better repeatability in that MCMC outputs were often more variable. We stress that the calculation of the likelihood function of those models has a non-negligible cost. We used here an IS approximation as in Stephens & Donnelly (2000) and the cost of this approximation increases considerably with the number of simulated gene trees. This type of model is then adequate for the adoption and the development of the AMIS algorithm: all particles simulated during the process are recycled, which minimizes the number of calls to the likelihood function. Due to this recycling process, the AMIS algorithm cannot be easily compared with other AIS schemes since those do not naturally involve any recycling step and since the natural mixture of importance samples is fraught with dangers, as explained at the beginning of this paper.

8. Conclusion

We have investigated in this paper an AIS method that extends the scope of the original deterministic multiple mixture approach of Owen & Zhou (2000) in that the sequence of importance proposals sequentially builds on the samples produced so far. The generality of the AMIS algorithm is that it offers a super-efficiency compared with other AIS techniques by allowing for an integral recycling of the past simulations. It thus provides a scope for processing those heterogeneous simulations as a whole and for treating the computing cost $\sum_{t=0}^{T} N_t$ as a single entity. The challenging issue of the theoretical convergence of the AMIS algorithm has not been solved in this paper, and the most promising direction in this respect is to derive acceptable growth rates for the sizes N_t when t goes to infinity.

References

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

Biernacki, C., Celeux, G. & Govaert, G. (2000). Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**, 719–725.

Biernacki, C., Celeux, G., Govaert, G. & Langrognet, F. (2006). Model-based cluster and discriminant analysis with the mixmod software. *Comput. Statist. Data Anal.* **51**, 587–600.

Cappé, O., Douc, R., Guillin, A., Marin, J.-M. & Robert, C. (2008). Adaptive importance sampling in general mixture classes. *Statist. Comput.* **18**, 587–600.

Cappé, O., Guillin, A., Marin, J.-M. & Robert, C. (2004). Population Monte Carlo. *J. Comput. Graph. Statist.* **13**, 907–929.

Chopin, N. (2002). A sequential particle filter method for static models. *Biometrika* **89**, 539–552.

Cornuet, J.-M. & Beaumont, M. (2007). A note on the accuracy of PAC-likelihood with microsatellite data. *Theor. Popul. Biol.* **71**, 12–19.

Cornuet, J.-M., Santos, F., Beaumont, M., Robert, C., Marin, J.-M., Balding, D., Guillemaud, T. & Estoup, A. (2008). Infering population history with *DIYABC*: a user-friendly approach to Approximate Bayesian Computation. *Bioinformatics* **24**, 2713–2719.

Del Moral, P., Doucet, A. & Jasra, A. (2006). Sequential Monte Carlo samplers. *J. Roy. Statist. Soc. Ser.* B, **68**: 411–436.

Douc, R., Guillin, A., Marin, J.-M. & Robert, C. (2007a). Convergence of adaptive mixtures of importance sampling schemes. *Ann. Statist.* **35**, 420–448.

Douc, R., Guillin, A., Marin, J.-M. & Robert, C. (2007b). Minimum variance importance sampling via Population Monte Carlo. *ESAIM: Probab. Stat.* **11**, 427–447.

Douc, R. & Moulines, E. (2008). Limit theorems for weighted samples with applications to sequential Monte Carlo methods. *Ann. Statist.* **36**, 2344–2376.

Doucet, A., de Freitas, N. & Gordon, N. (2001). *Sequential Monte Carlo methods in practice*. Springer-Verlag, New York.

Doucet, A., Godsill, S. & Andrieu, C. (2000). On sequential Monte Carlo sampling methods for Bayesian filtering. *Statist. Comput.* **10**, 197–208.

Gordon, N., Salmond, J. & Smith, A. (1993). A novel approach to non-linear/non-Gaussian Bayesian state estimation. *IEEE Proc. Radar and Signal Process.* **140**, 107–113.

Haario, H., Saksman, E. & Tamminen, J. (1999). Adaptive proposal distribution for random walk Metropolis algorithm. *Comput. Statist.* **14**, 375–395.

Haario, H., Saksman, E. & Tamminen, J. (2001). An adaptive Metropolis algorithm. *Bernoulli* **7**, 223–242.

Hesterberg, T. (1995). Weighted average importance sampling and defensive mixture distributions. *Technometrics* **37**, 185–194.

Hey, J. & Nielsen, R. (2004). Multilocus methods for estimating population sizes, migration rates and divergence time, with applications to the divergence of Drosophila pseudoobscura and D. persimilis. *Genetics* **167**, 747–760.

Liu, J. (2001). *Monte Carlo strategies in scientific computing*. Springer-Verlag, New York.

Liu, J., Liang, F. & Wong, W. (2001). A theory of dynamic weighting in Monte Carlo computation. *J. Amer. Statist. Assoc.* **96**, 561–573.

Nelder, J. & Mead, R. (1965). A simplex method for function minimization. *Comput. J.* **7**, 308–313.

Ortiz, L. & Kaelbling, L. (2000). Adaptive importance sampling for estimation in struc- tured domains. In *Proceedings of the Sixteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-2000)*, 446–454. Morgan Kaufmann Publishers, San Francisco, CA.

Owen, A. & Zhou, Y. (2000). Safe and effective importance sampling. *J. Amer. Statist. Assoc.* **95**, 135–143.

Peel, D. & McLachlan, G. (2000). Robust mixture modelling using the *t* distribution. *Statist. Comput.* **10**, 339–348.

Pennanen, T. & Koivu, M. (2004). An adaptive importance sampling technique. In *Monte Carlo and Quasi-Monte Carlo methods 2004* (eds H. Niederreiter & D. Talay), 443–455. Springer-Verlag, Berlin.

Raftery, A. & Bao, L. (2010). Estimating and projecting trends in HIV/AIDS generalized epidemics using Incremental Mixture Importance Sampling. *Biometrics* **66**, 1162–1173.

Ripley, B. D. (1987). *Stochastic simulation*. John Wiley & Sons Inc., New York.

Robert, C. & Casella, G. (2004). *Monte Carlo statistical methods* 2nd edn. Springer-Verlag, New York.

Rubinstein, R. & Kroese, D. (2004). *The cross-entropy method. A unified approach to combinatorial optimization, Monte-Carlo simulation, and machine learning*. Springer-Verlag, New York.

Sirén, J., Marttinen, P. & Corander, J. (2010). Reconstructing population histories from single-nucleotide polymorphism data. *Mol. Biol. Evol.* **28**, 673–683.

Stephens, M. & Donnelly, P. (2000). Inference in molecular population genetics (with discussion). *J. Roy. Statist. Soc. Ser. B* **62**, 605–655.

Veach, E. & Guibas, L. (1995). Optimally combining sampling techniques for Monte Carlo rendering. In *SIGGRAPH '95 Conference Proceedings*, 419–428. Addison-Wesley, Reading, MA.

Jean-Michel Marin, Institut de Mathématiques et Modélisation de Montpellier (UMR CNRS 5149), Université Montpellier 2, Montpellier, France.
E-mail: jean-michel.marin@univ-montp2.fr