



Adaptive importance sampling in signal processing



Mónica F. Bugallo^{a,*}, Luca Martino^b, Jukka Corander^b

^a Department of Electrical and Computer Engineering, Stony Brook University, USA

^b Department of Mathematics and Statistics, University of Helsinki, Helsinki, Finland

ARTICLE INFO

Article history:

Available online 22 June 2015

Keywords:

Adaptive importance sampling
Markov chain Monte Carlo
Population Monte Carlo
Particle filtering

ABSTRACT

In Bayesian signal processing, all the information about the unknowns of interest is contained in their posterior distributions. The unknowns can be parameters of a model, or a model and its parameters. In many important problems, these distributions are impossible to obtain in analytical form. An alternative is to generate their approximations by Monte Carlo-based methods like Markov chain Monte Carlo (MCMC) sampling, adaptive importance sampling (AIS) or particle filtering (PF). While MCMC sampling and PF have received considerable attention in the literature and are reasonably well understood, the AIS methodology remains relatively unexplored. This article reviews the basics of AIS as well as provides a comprehensive survey of the state-of-the-art of the topic. Some of its most relevant implementations are revisited and compared through computer simulation examples.

© 2015 Elsevier Inc. All rights reserved.

1. Introduction

The past two decades have witnessed an astounding growth in the field of statistical signal and information processing. The advances in the field have been particularly compelling in terms of computational methods. This progress has been unleashed with the ever increasing computing power available nowadays at very low costs. Bayesian signal and information processing has perhaps benefited most significantly from the advances in computing hardware and algorithms [1]. With this technological momentum, the Bayesians freed themselves from the computational limitations of the classical approaches and found a range of new opportunities for handling models of very high complexities. As a result, Bayesian theory and practice have seen remarkable advances. Many disciplines have benefited from these developments including engineering, bioinformatics, econometrics, astronomy, climatology, and computational physics.

A driving force for the increased popularity of Bayesian methods has been the progress in the theory and practice of reject-accept methods [2,3], Markov chain Monte Carlo (MCMC) algorithms [4–6] and particle filtering (PF) [7]. All these methodologies are based on approximating distributions of unknown parameters (or states) of interest by samples and associated weights [8–12]. Recently, adaptive importance sampling (AIS) [13] has resurfaced as an alternative to MCMC sampling and it has gained attention in various research communities [14]. Unlike MCMC sampling, which

exploits Markov chains to reach a desired (target) distribution for generating the desired samples, the AIS methodology draws them from distributions that improve with iterations. AIS schemes do not reject any of the generated samples, do not necessitate of burn-in periods and can be implemented easily [15]. Additionally, with AIS methods, it is possible to easily estimate the normalizing constant of the posterior target distribution (also called Bayesian evidence, marginal likelihood or partition function) by averaging the unnormalized importance weights. The marginal likelihood is particularly useful in model selection [2,16]. Furthermore, in general, the consistency of AIS schemes is easily guaranteed by weak basic conditions, whereas the theoretical study of the convergence of a Markov chain generated by an adaptive MCMC method needs more careful considerations [17–19].

The key idea behind AIS is importance sampling [20,21]. It boils down to generating samples from a cleverly selected distribution called instrumental, importance or proposal function. This distribution is different from the target distribution because one assumes that direct sampling from the target distribution is infeasible. Good proposal functions do closely resemble the target distribution. Once the samples are generated from a proposal function, they are assigned weights. The samples with their weights represent an approximation of the target distribution. The key problem is that good proposal functions are very hard to choose in an automated fashion. On the other hand, Bayesian theory suggests that one can start by generating samples from the prior and weighting them using the posterior (usually the target distribution). The obtained set of samples and weights is an approximation of the posterior that can be improved. To that end, a tempting idea is to

* Corresponding author.

E-mail address: monica.bugallo@stonybrook.edu (M.F. Bugallo).

use this approximation in constructing a better instrumental function than the originally used prior. If this works, the samples and weights of the improved proposal function will produce a better approximation of the posterior. Consequently, this approximation can also serve to again construct a better proposal function. This is naturally the case with all the subsequent approximations too. Thus, the importance sampling keeps adapting as one proceeds with iterations, which is why the methodology is referred to as AIS. In this process, learning takes place from samples and weights obtained in previous iterations.

This paper reviews the theory and practice of AIS when applied to problems in signal processing. More specifically, the next section provides the general background and discusses the basics of the AIS method. After that, an extensive survey of the literature on the topic is presented and includes a summary of the most important convergence results. The section on implementations details some of the most popular AIS algorithms including the population Monte Carlo (PMC) [15], the AIS with mixture Gaussian proposals [22], the adaptive multiple importance sampling (AMIS) [14,23], and the adaptive population importance sampler (APIS) [24,25]. The section on examples analyzes and compares the discussed algorithms as well as the MCMC through computer simulations. Three relevant problems are examined: the well known “banana-shaped” distribution [17] as proof of concept, a high-dimensional setting with a multimodal target distribution, and an autoregressive (AR) model in a non-Gaussian noise scenario. The final section provides some closing remarks.

2. Background and the basics of AIS

A very important challenge in many signal processing problems is the reliable approximation of target distributions. Unfortunately in many scenarios such approximation cannot be obtained analytically and one has to resort to numerical methods. A popular class of computational algorithms is the Monte Carlo family, which is particularly suitable to the broad and challenging problems of non-linear and non-Gaussian nature. The target distribution can be, for example, the posterior distribution of the unknowns or a predictive distribution of observations. Within the Monte Carlo-based methods, MCMC algorithms use correlated samples to approximate a target distribution [26–28]. MCMC techniques yield a Markov chain with a stationary distribution that coincides with the target. Thus, after a sufficient amount of iterations, the MCMC methods approximate the target distribution with a random measure composed of generated samples with equal weights. In adaptive MCMC sampling, one also uses proposal functions and they may depend on already generated samples. The AIS has the same objective as MCMC but it operates with samples whose weights are all different. The difference in weights is due to the fact that with AIS the generated samples are always used in the estimation and are drawn by a proposal function different from the desired distribution. Some proposal functions are better than others and the main challenge lies in the construction of an appropriate proposal. With AIS, the aim is to implement importance sampling in an iterative manner, where one exploits the samples and weights of past iterations in constructing improved importance functions.

PF is another Monte Carlo methodology that uses importance sampling [29]. With PF, the main objective is to track in time various dynamic distributions of interest [7,30,31]. As pointed out, PF, like AIS, exploits importance sampling, but does not usually take advantage of iterative learning while it tracks its target distributions. Thus, in a nutshell, AIS shares important features with both MCMC sampling and PF. Like in adaptive MCMC sampling, it uses previous samples to construct better proposal functions, and like PF, it employs importance sampling to avoid generation of samples from infeasible distributions. Another advantage of the AIS

approach (w.r.t. the MCMC techniques) is that it is easier to design schemes using a population of different proposal distributions (sharing information by the weighted samples).

The mathematical basics of AIS can be described as follows. Suppose that we want to approximate a target distribution $p(x)$ by a set of samples and weights. Here $x \in \mathbb{R}^{d_x}$ is the unknown state of the system of dimension d_x . This distribution is most often a posterior, $p(x | y_{1:N_y})$, where $y_n \in \mathbb{R}^{d_y}$ represents observations with information about x , and $y_{1:N_y} \equiv \{y_1, y_2, \dots, y_{N_y}\}$, with N_y indicating the number of available observations.

First, we review the concept of importance sampling [20]. Let the samples used for approximation be drawn from $p(x)$ itself and let them be denoted by $x^{(m)}$, $m = 1, 2, \dots, M$, with M representing the number of samples that are generated. Then

$$p^M(x) = \sum_{m=1}^M \frac{1}{M} \delta(x - x^{(m)}), \quad (1)$$

where the approximating distribution, $p^M(x)$, is discrete and $\delta(x - x^{(m)})$ is the unit delta measure concentrated at $x^{(m)}$. When it is difficult or impossible to draw samples from $p(x)$, the alternative is to use a proposal function $\pi(x)$, with a shape as close as possible to $p(x)$ and support enclosing that of $p(x)$. If $x_0^{(m)} \sim \pi_0^{(m)}(x)$ (where the subscript denotes the iteration), then the approximation of $p(x)$ is

$$p_0^M(x) = \sum_{m=1}^M w_0^{(m)} \delta(x - x_0^{(m)}), \quad (2)$$

where $w_0^{(m)}$ is the normalized importance weight obtained from

$$w_0^{(m)} = \frac{\bar{w}_0^{(m)}}{\sum_{j=1}^M \bar{w}_0^{(j)}}, \quad (3)$$

where

$$\bar{w}_0^{(m)} = \frac{p(x_0^{(m)})}{\pi_0^{(m)}(x_0^{(m)})}. \quad (4)$$

The generation of samples from a proposal function and the assignment of weights is known as importance sampling. We note that the superscript in the proposal function indicates that for each sample generated, one can use a different proposal. The nature of this *difference* of proposal probability density functions (pdfs) can be in the type of distribution (e.g., some samples could be generated from a Gaussian, some from a mixture Gaussian, etc.) or in the parameters of the proposal (e.g., assuming a simple case where all samples are generated from Gaussians, those Gaussians could have different parameters depending on the particular sample.) In this section, for simplicity in notation, we relate the proposal distribution directly to the unknown variable of the system x . In general, and as seen in the next sections, the proposal may not only relate to the variable x but to other parameters.

The idea behind AIS is to use the random measure $\chi_0 = \{x_0^{(m)}, w_0^{(m)}\}_{m=1}^M$ for creating a better proposal function than $\pi_0(x)$ (see Fig. 1 for a graphical description of the *modus operandi* of AIS in two iterations). If this new proposal function is $\pi_1(x)$, and it is used to obtain a new random measure $\chi_1 = \{x_1^{(m)}, w_1^{(m)}\}_{m=1}^M$, then the approximation of $p(x)$ with the new set of samples is

$$p_1^M(x) = \sum_{m=1}^M w_1^{(m)} \delta(x - x_1^{(m)}), \quad (5)$$

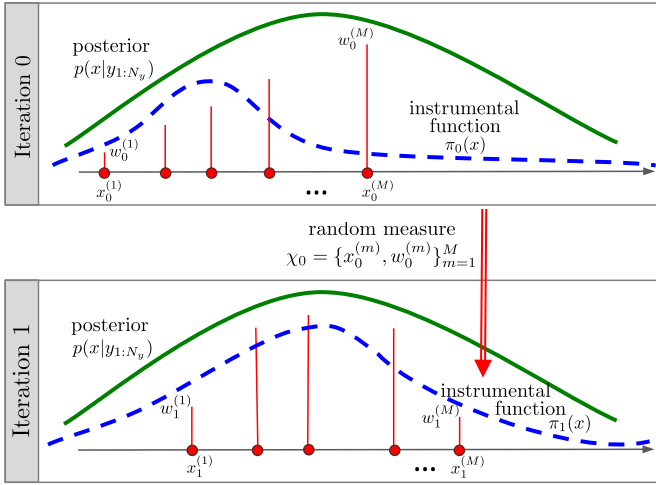


Fig. 1. The idea behind AIS with two iterations.

where the subscript of $p_1^M(x)$ indicates the approximation of $p(x)$ at the first iteration. One can also combine the initial random measure χ_0 with χ_1 to improve the approximation (5), e.g., by

$$\tilde{p}_1^M(x) = \sum_{m=1}^M \left(\tilde{w}_0^{(m)} \delta(x - x_0^{(m)}) + \tilde{w}_1^{(m)} \delta(x - x_1^{(m)}) \right), \quad (6)$$

where the new weights $\tilde{w}_0^{(m)}$ and $\tilde{w}_1^{(m)}$ are appropriately recomputed so that they all sum up to one, namely,

$$\tilde{w}_0^{(m)} = \frac{\tilde{w}_0^{(m)}}{\sum_{j=1}^M \tilde{w}_0^{(j)} + \sum_{j=1}^M \tilde{w}_1^{(j)}}, \quad \text{and}$$

$$\tilde{w}_1^{(m)} = \frac{\tilde{w}_1^{(m)}}{\sum_{j=1}^M \tilde{w}_0^{(j)} + \sum_{j=1}^M \tilde{w}_1^{(j)}}. \quad (7)$$

The process can obviously continue iteratively (in the sequel, the subscript i will denote iteration number and the symbol I will indicate the maximum number of iterations). In the second iteration, one can use the random measure χ_1 and create yet an even better proposal function. Or, one can use both measures $\chi_{0:1}$ (here, $\chi_{0:1} \equiv \{\chi_0, \chi_1\}$) to obtain the new proposal function. The iterative process proceeds until a stopping condition is met. It is important to note that the samples from all the iterations can be used once their weights are properly computed. Finally, it is important to remark that

$$\hat{z} = \frac{1}{2M} \left(\sum_{j=1}^M \tilde{w}_0^{(j)} + \sum_{j=1}^M \tilde{w}_1^{(j)} \right) \approx \int_{\mathbb{R}^{d_x}} p(x) dx, \quad (8)$$

is an estimator of the normalizing constant of $p(x)$.

3. State-of-the-art

The idea of importance sampling was proposed for the first time more than 50 years ago [32]. It was introduced in Bayesian inference in [33] and later developed in [34,35]. The concept of learning about the target distribution along with Monte Carlo sampling was proposed in [36,37], and more recent efforts include [22,38]. It is important to point out that learning from past samples has been explored in the MCMC sampling literature, where the main motivation has primarily been the development of MCMC methods for the local trap problem [27].

A resurgence of interest in AIS schemes happened after the publication of the population Monte Carlo (PMC) sampling method

in [15,28]. It is based on representing the proposal functions by mixtures of kernels. The underlying idea is simple and flexible and several variants have been proposed. In particular, two general versions of PMC aiming at complete adaptation of the proposal functions, have received special attention. In one, the proposal functions are fit to the target distribution so that they minimize the variance or the Kullback–Leibler divergence [39]. In the other method, one updates the weights and parameters of the mixture distribution by using an entropy criterion [22]. In [40], a PMC method was applied to missing data problems, where a sequence of importance functions dependent on both the iteration and sample index was proposed. Additional applications of PMC sampling include the analysis of ion channels using a fixed dimension model [41], interpretation of static images in a robot vision setup [42], analysis of complex traits for phenotypic data interpretation of pedigreed populations [43], motif discovery for deciphering genetic regulatory codes [44], and investigation of detection systems for biological threats [45]. Several other related works can be found in the literature [46–48]. For instance, in [48], the loss of diversity in the population due to the resampling step is reduced by artificially forcing a pre-defined amount of the highest importance weights to be equal.

Other variants of AIS schemes have been presented in the literature. Some approaches focus on the use of nonparametric mixture distributions as proposal functions [49–52]. The number of components in the mixture varies with iterations according to certain statistical criteria. In many of these schemes, the new added mixands (i.e., a new component) are located and weighted by using the samples generated at the previous iteration [49,50]. Then, certain components are removed or replaced according to a resampling step. The parameters of the components are adapted using global [49] or local [50] strategies. In other approaches, the optimal parameters of the components are obtained via empirical estimation [51]. In [53], a strategy that initiates with a very large number of mixands is described. As the algorithm proceeds, some of the mixands are merged and some are removed using a clustering procedure. An alternative to this method is the incremental mixture importance sampling (IMIS) method. With IMIS, the proposal mixture distribution is adapted by incrementally adding components [52] based on the highest importance weights from the previous iteration.

More recently, the idea of incremental mixtures has been promoted by another AIS scheme, the adaptive multiple importance sampling (AMIS) method [14,23,54]. In AMIS, at a particular iteration, only one proposal function is used to generate all samples. However, at each iteration the past and present weights are recomputed by using the so-called deterministic mixture approach [55,56]. The mixture is composed by the present and all past proposal distributions. Thus, a new component is incorporated in the mixture at each iteration (in this sense, it is incremental). Also, the parameters of the next iteration proposal are obtained by using all particles and weights up to that moment. It has been demonstrated that the deterministic mixture approach stabilizes the iterative importance sampling estimator. Thus, AMIS often provides better performance than other techniques. For instance, it has been successfully applied to biological data analysis [57].

In the previous works, some elements of the mixture, e.g., weights or covariances, remain fixed, i.e., are not adapted. A complete adaptation of a mixture of proposal distributions has been designed in [22], from the theoretical and practical point of view. In this case, the weights and parameters of each mixand are updated in order to minimize the Kullback–Leibler divergence between the target and proposal distributions. Since the adaptation of weights and covariance matrices is in general critical for adequate performance of the algorithms, other methods only adapt the location parameters [24], in order to provide a more robust

Table 1

Summary of the notation of the AIS methodology.

Indices	Description	Functions	Description
i	Index denoting the iteration of the algorithm	p	Target function
I	Total number of iterations	π	Proposal function
m	Index denoting the m -th sample	q	Mixand in the proposal π
M	Total number of samples at each iteration	$\bar{w}_i^{(m)}$	Unnormalized weight of the m -th sample at the i -th iteration
		$\tilde{w}_i^{(m)}$	Unnormalized weight used in APIS
d	Index denoting the d -th mixand	$w_i^{(m)}$	Weight of the m -th sample, normalized considering the M samples drawn at the i -th iteration
D	Total number of mixands in the proposal π	$\tilde{w}_\tau^{(m)}$	Weight of the m -th sample at the τ -th iteration ($\tau \leq i$), normalized considering all the samples drawn up to the i -th iteration

performance. With the adaptive population importance sampling (APIS) method [24,25], each component of the mixture can use different parameters as in PMC, but the parameters of a mixand are updated to reflect the local features of the target. The weights of the mixture remain the same, equal for all the mixands in its basic formulation (away from the 0 and 1 values, as suggested in [58]). APIS also takes advantage of the deterministic mixture approach [55,56] but with the population of proposal distributions at the same iteration, instead of creating a mixture among proposal distributions obtained at different time steps, as in AMIS. Finally, combinations of MCMC and importance sampling techniques have been studied to obtain further improved AIS methods. For instance, MCMC steps are used to accelerate the adaptation of the AIS technique [25,59] or, more generally, the MCMC outputs are used to build a proposal distribution for AIS estimation [60].

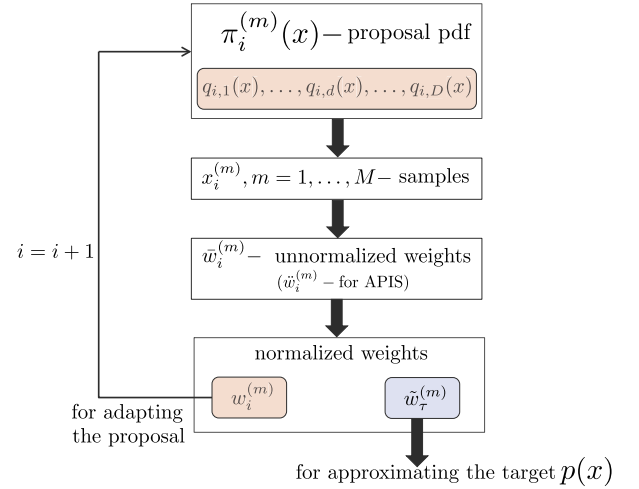
In general, the proof of consistency of the estimators obtained by the AIS schemes is based on the same arguments used for the standard importance sampling method [2,16]. In fact, the AIS estimator that is calculated at the last iteration of the method (I is the maximum number of iterations of the algorithm) can be interpreted as a standard-static importance sampling estimator built using I different proposal distributions. The details of this analysis can be found in [25] and Chapter 14 of [2]. The consistency of the method when the number of samples tend to infinity for a fixed number of iterations ($M \rightarrow \infty$ with fixed I) is, in general, straightforward based on simple Monte Carlo arguments. However, the proof for an infinite number of iterations and a fixed number of samples ($I \rightarrow \infty$ with fixed M) is more involved since the partial estimators at each iteration of the AIS scheme are usually biased (see discussion in [25]). The convergence of some specific AIS schemes like the AMIS have received special attention in [23]. There, the algorithm, which originally considered all samples from the past and current iterations for update of the proposal, was modified to facilitate the theoretical analysis and only the current samples at a particular iteration were used in the update of the proposal function. Some additional theoretical results can be found in [30,39].

4. AIS schemes

In this section, we first review the generic characteristics of the AIS methodology as well as its basic steps. Then, we provide details of four different implementations of AIS. In particular, we focus on the standard PMC algorithm [15], the AIS with mixture Gaussian proposals [22], the adaptive multiple importance sampling (AMIS) [14,23], and the adaptive population importance sampler (APIS) [24,25].

4.1. Outline of the general scheme

As indicated in Section 2, the objective of the AIS algorithms is the approximation of a target distribution, $p(x)$, by a set of samples and weights. Since the samples cannot generally be drawn directly

**Fig. 2.** Outline of the general AIS scheme with its main steps and the used notation.

from $p(x)$, one uses a proposal function, $\pi(x)$, to generate the samples and assigns them weights, which indicate the relevance of the samples to the approximation. The AIS methods are iterative and improve the proposal function with each iteration. We note that all methods reviewed in this paper assume parametric proposal functions. Alternative schemes could be considered as is the case in [49–52], where nonparametric proposal pdfs are used. The main notation is summarized in Table 1 and a graphical representation of the general outline of the AIS methodology including the notation is provided in Fig. 2.

In general, at the i -th iteration, the AIS techniques, move from a random measure, $\chi_{i-1} = \{x_{i-1}^{(m)}, w_{i-1}^{(m)}\}_{m=1}^M$, to a new random measure, χ_i , where the samples are generated from an improved proposal constructed from χ_{i-1} and the weights are properly updated. In particular, the generation of new samples is given by

$$x_i^{(m)} \sim \pi_i^{(m)}, \quad m = 1, \dots, M, \quad (9)$$

where $\pi_i^{(m)}$ is the proposal function for the m -th sample at the i -th iteration. It is in the construction of the proposal where the AIS algorithms are differentiated:

- **Standard PMC [15]:** Each sample is generated from a different proposal of parameters updated from the previous iteration and defined by the previous sample, i.e., $x_i^{(m)} \sim \pi_i^{(m)}(x) = q_{i,m}(x | \mu_{i-1}^{(m)}, \Sigma_{i-1}^{(m)})$, where $\mu_{i-1}^{(m)}$ and $\Sigma_{i-1}^{(m)}$ denote the mean and covariance matrix defining the parametric distribution. Note that one can interpret that PMC uses a mixture of distributions $q_{i,m}$, $i = 0, \dots, I$, $m = 1, \dots, M$, where the selection of a particular mixand to draw one sample is deterministic (one sample is generated from each of the M proposal pdfs). This observation establishes an important connection with the rest of techniques analyzed below.

- **AIS with mixture Gaussian proposals [22]:** There is one proposal distribution from which all samples are generated. This function is a mixture Gaussian with D components of parameters updated at each iteration. The m -th sample at the i -th iteration is generated by randomly selecting one of the mixands, i.e., $x_i^{(m)} \sim \pi_i^{(m)}(x) = \sum_{d=1}^D \alpha_{i-1,d} q_{i,d}(x | \mu_{i-1,d}, \Sigma_{i-1,d})$, where $\alpha_{i-1,d}$, $\mu_{i-1,d}$ and $\Sigma_{i-1,d}$ denote the weight, mean and covariance matrix of the d -th component of the mixture. In the sequel, we also refer to this method as Mix-AIS.
- **AMIS [14,23]:** The M samples, at the i -th iteration, are generated using the same proposal function with parameters obtained from all samples and weights up to that iteration, i.e., $x_i^{(m)} \sim \pi_i^{(m)}(x) = q_i(x | \mu_{i-1}, \Sigma_{i-1})$, where μ_{i-1} and Σ_{i-1} denote the mean and covariance matrix calculated at the $(i-1)$ -th iteration using all samples and weights. AMIS also exploits the deterministic mixture approach in the computation of the weights. At the i -th iteration, all iM samples up to that iteration, have been generated from the temporal mixture $\frac{1}{i-1} \sum_{j=0}^{i-1} q_j(x | \mu_{j-1}, \Sigma_{j-1})$.
- **APIS [24,25]:** At each iteration, there are D proposal pdfs and the same number of samples is drawn from each of them, i.e., if one generates M samples, $N = \frac{M}{D} \geq 2$ samples come from each of the proposal functions. Therefore, $x_i^{(m)} \sim \pi_i^{(m)}(x) = q_{i,d}(x | \mu_{i-1,d}, \Sigma_{i-1,d})$, $d = \lfloor \frac{m-1}{N} \rfloor + 1$, where $\mu_{i-1,d}$ and $\Sigma_{i-1,d}$ denote the mean and covariance matrix of the d -th proposal and are updated using the samples generated from that proposal. A useful interpretation is that APIS uses a mixture of proposal distributions, $q_{i,d}$, where the selection of the mixand to draw the sample from is deterministic (N samples are generated from each of the D proposal pdfs such that $ND = M$).

For each of the previous variants of AIS, one needs to update the corresponding weights taking into account the considered proposal pdf. In the next subsections we detail the specifics of each of the approaches. The main steps of all the schemes are:

Initialization: Obtain $\chi_0 = \{x_0^{(m)}, w_0^{(m)}\}_{m=1}^M$, from a prior distribution. This step also involves initialization of the parameters needed for the first proposal.

For $i = 1 : I$

1. **Generate new samples:** $x_i^{(m)} \sim \pi_i^{(m)}(x)$, $m = 1, \dots, M$.
The way the proposal functions $\pi_i^{(m)}(x)$ are built distinguish the different approaches.
2. **Calculate the unnormalized weights:**

$$\tilde{w}_i^{(m)} = \frac{p(x_i^{(m)})}{\pi_i^{(m)}(x_i^{(m)})}, \quad m = 1, \dots, M. \quad (10)$$

As seen later, the unnormalized weights are usually calculated as the ratio between the target and the proposal. The AMIS algorithm evaluates the weights by considering the cumulative proposal resulting from the addition of all proposal pdfs up to the current iteration. In addition, it also recalculates the weights of samples from previous iterations in the same way. The APIS algorithm contemplates two different unnormalized weights: on one hand, the ratio between the target and the proposal is considered, and on the other hand, the ratio between the target and the composite proposal resulting from the mixture of all individual proposals is calculated. The former is used for adaptation of the next iteration parameters of each proposal pdf, while the later is used for approximation of the target distribution.

Table 2

Expressions of the proposal and number of mixands per proposal for the different AIS implementations.

Method	Proposal pdfs	D
PMC	$\pi_i^{(m)}(x) = q_{i,m}(x \mu_{i-1}, \Sigma_{i-1})$	$D = M$
Mix-AIS	$\pi_i^{(m)}(x) = \sum_{d=1}^D \alpha_{i-1,d} q_{i,d}(x \mu_{i-1,d}, \Sigma_{i-1,d})$	D
AMIS	$\pi_i^{(m)}(x) = q_i(x \mu_{i-1}, \Sigma_{i-1})$	$D = 1$
APIS	$\pi_i^{(m)}(x) = q_{i,d}(x \mu_{i-1,d}, \Sigma_{i-1,d})$, $d = \lfloor \frac{m-1}{N} \rfloor + 1$	$D = \frac{M}{N}$

3. **Normalize the weights:** Two different types of normalization are considered:

- (a) For adaptation of the necessary parameters for the next iteration proposal

$$w_i^{(m)} = \frac{\tilde{w}_i^{(m)}}{\sum_{j=1}^M \tilde{w}_i^{(j)}}, \quad m = 1, \dots, M. \quad (11)$$

The standard PMC also uses these weights for the resampling operation. In PMC the discrete random measure degenerates quickly and only few samples are assigned meaningful weights. This degradation leads to a deteriorated performance of the method. Resampling eliminates sampling with small weights and replicates particles with large weights [2,16,61,62]. It is important to note that the AMIS approach does not normalize the weights using the previous expression but the next one. This is due to the cumulative nature of the algorithm, which constructs the proposal pdfs by updating the parameters with the samples from all previous iterations. Therefore, the weights used for adaptation of these parameters are normalized through all the iterations.

Using the generated samples and the calculated weights, one obtains the next random measure, $\chi_i = \{x_i^{(m)}, w_i^{(m)}\}_{m=1}^M$.

- (b) For approximation of the target distribution as well as calculation of point estimates of unknowns

$$\tilde{w}_\tau^{(m)} = \frac{\tilde{w}_\tau^{(m)}}{\sum_{\rho=0}^i \sum_{j=1}^M \tilde{w}_\rho^{(j)}}, \quad \tau = 0, \dots, i, \quad m = 1, \dots, M. \quad (12)$$

This is a global normalization across all weights from previous and current iterations.

4. **Approximate the target distribution:** To achieve this, one uses all samples and globally normalized weights up to the present iteration,

$$\tilde{p}_i^M(x) = \sum_{m=1}^M \left(\tilde{w}_0^{(m)} \delta(x - x_0^{(m)}) + \tilde{w}_1^{(m)} \delta(x - x_1^{(m)}) + \dots + \tilde{w}_i^{(m)} \delta(x - x_i^{(m)}) \right). \quad (13)$$

5. **Adapt the parameters for the next proposal:** Depending on the version of the AIS method, one has to update the parameters for the next iteration proposal accordingly. Details of this adaptation are provided in the next subsections.

Fig. 2 illustrates the general AIS scheme and Table 2 compares the expressions of the proposal pdfs, $\pi_i^{(m)}$, as well as the number of mixands in the proposal pdfs used by each of the considered AIS implementations.

4.2. Standard PMC

Here we detail the original PMC sampling algorithm proposed in [15]. We consider that at the beginning of the i -th iteration, the random measure from the $(i-1)$ -th iteration, $\chi_{i-1} = \{x_{i-1}^{(m)}, w_{i-1}^{(m)}\}_{m=1}^M$, is available. To obtain the new random measure, the m -th sample is propagated from a proposal distribution, $\pi_i^{(m)}(\cdot)$, described by a set of parameters obtained from m -th sample at the previous iteration, i.e., $\pi_i^{(m)}(x) = q_{i,m}(x_i^{(m)} | \mu_{i-1}^{(m)}, \Sigma_{i-1}^{(m)})$. This implies that at each iteration there are M different proposal pdfs. We note that as described in [15] all the M proposals are of the same distribution type, e.g., all Gaussians or all Student's t -distributions. However, the parameters of those proposal pdfs are different in that they come from different parental samples. We also note that the proposals do not change their type from iteration to iteration. In a more general setting, one could have the M proposal pdfs being of different distribution types and changing their types at each iteration. The proposals considered in [15] are characterized by their means and covariances matrices, i.e., the i -th proposal depends on $\mu_{i-1}^{(m)}$ and $\Sigma_{i-1}^{(m)}$ from the previous iteration.

Table 3 summarizes the algorithm. Note that, at each iteration, the M samples obtained from the PMC method can be interpreted to come from a mixture $\frac{1}{M} \sum_{m=1}^M q_{i,m}(x | \mu_{i-1}^{(m)}, \Sigma_{i-1}^{(m)})$, where exactly one sample is generated from each of the mixands. However, unlike AMIS and APIS, PMC does not take advantage of this consideration. This is perceived in the calculation of the weights since the corresponding weight for each sample only reflects the proposal that the sample was drawn from and not the entire mixture.

4.3. AIS with mixture Gaussian proposals

In the PMC sampling algorithm described in [22], the proposal distribution at the i -th iteration is the same for all particles and is given by a mixture of D Gaussians, i.e., $\pi_i^{(m)}(x) = \sum_{d=1}^D \alpha_{i-1,d} q_{i,d}(x | \mu_{i-1,d}, \Sigma_{i-1,d})$, where $q_{i,d}(x | \mu_{i-1,d}, \Sigma_{i-1,d}) = \mathcal{N}(x | \mu_{i-1,d}, \Sigma_{i-1,d})$ is the d -th mixand whose parameters as well as its weight in the mixture are updated at each iteration by using the generated samples and calculated weights. The generation of samples at the i -th iteration involves the selection of a component of the mixture according to the weights $\alpha_{i-1,d}$, $d = 1, \dots, D$. A sort of resampling is therefore inherent to the generation step. Table 4 summarizes the algorithm. We note that the subscript -1 used for initialization of the algorithm refers to the parameters of the prior distribution used to obtain the first set of samples. It is important to remark that the adaptation of the parameters in Table 4 is only valid if all the mixture components are Gaussians. In other cases, an additional theoretical derivation is needed [22].

4.4. AMIS

In the AMIS approach [14,23] the proposal functions are incremental in the sense that at each iteration the new proposal is characterized by parameters that are obtained using all samples and weights from previous iterations. This proposal is used for generation of all samples, i.e., $\pi_i^{(m)}(x) = q_i(x | \mu_{i-1}, \Sigma_{i-1})$, $\forall m$. Also, at each iteration the weights of all samples (current and past) are computed (or recomputed in the case of past samples) taking into account the incremental nature of the proposal, i.e., the weights measure the adequacy of the samples with respect to the composite distribution that accounts for all iterations proposal pdfs up

Table 3

Standard PMC.

Assume: $\chi_0 = \{x_0^{(m)}, w_0^{(m)} = \frac{1}{M}\}_{m=1}^M$, where $x_0^{(m)}$, $m = 1, \dots, M$ were obtained from a prior distribution.

Set: $\mu_0^{(m)} = x_0^{(m)}$ $m = 1, \dots, M$.

$\Sigma_0^{(m)} = \Sigma^{(m)}$ $m = 1, \dots, M$, where $\Sigma^{(m)}$ is the predefined covariance matrix of the prior.

For $i = 1, \dots, I$

1. Generate new samples: $x_i^{(m)} \sim \pi_i^{(m)}(x) = q_{i,m}(x | \mu_{i-1}^{(m)}, \Sigma_{i-1}^{(m)})$, $m = 1, \dots, M$.
2. Calculate the unnormalized importance weights:

$$\tilde{w}_i^{(m)} = \frac{p(x_i^{(m)})}{q_{i,m}(x_i^{(m)} | \mu_{i-1}^{(m)}, \Sigma_{i-1}^{(m)})}, \quad m = 1, \dots, M.$$

3. Normalize the weights:
 - (a) For resampling as well as adaptation of the necessary parameters for the next iteration proposal

$$w_i^{(m)} = \frac{\tilde{w}_i^{(m)}}{\sum_{j=1}^M \tilde{w}_i^{(j)}}, \quad m = 1, \dots, M.$$

Obtain: $\chi_i = \{x_i^{(m)}, w_i^{(m)}\}_{m=1}^M$.

- (b) For approximation of the target distribution as well as calculation of point estimates of unknowns

$$\tilde{w}_\tau^{(m)} = \frac{\tilde{w}_\tau^{(m)}}{\sum_{\rho=0}^i \sum_{j=1}^M \tilde{w}_\rho^{(j)}} \quad \tau = 0, \dots, i, \quad m = 1, \dots, M.$$

4. Approximate the target distribution:

$$\tilde{p}_i^M(x) = \sum_{m=1}^M \left(\tilde{w}_0^{(m)} \delta(x - x_0^{(m)}) + \tilde{w}_1^{(m)} \delta(x - x_1^{(m)}) + \dots + \tilde{w}_i^{(m)} \delta(x - x_i^{(m)}) \right).$$

5. Resample by replication of the most adequate samples according to the weights. Obtain a new set of samples $\{\tilde{x}_i^{(m)}\}_{m=1}^M$ for adaptation of the necessary parameters for the next iteration proposal.
6. Adapt the parameters for the next proposal:

$$\mu_i^{(m)} = \tilde{x}_i^{(m)}, \quad m = 1, \dots, M,$$

$$\Sigma_i^{(m)} = \Sigma_{i-1}^{(m)} \quad m = 1, \dots, M.$$

to that iteration, $\frac{1}{i+1} \sum_{j=0}^i q_j(x | \mu_{j-1}, \Sigma_{j-1})$. This mixture, with increasing number of components, can be considered as the complete reference proposal pdf for weight calculation used in AMIS. All the samples from all the iterations as well as the properly recomputed and normalized weights are used for adjustment of the parameters for the next iteration proposal. Table 5 summarizes the algorithm.

4.5. APIS

The algorithm proposed in [24,25] uses D proposal functions at each iteration to obtain the M samples, i.e., it generates $N = \frac{M}{D} \geq 2$ samples per proposal function ($N \in \mathbb{N}$). Namely, the total number of samples per iteration is $M = ND$. Since N samples are generated deterministically from each mixand at each iteration, one can also consider that the $M = ND$ samples drawn at the i -th iteration are distributed according to the mixture pdf $\frac{1}{D} \sum_{d=1}^D q_{i,d}(x | \mu_{i-1,d}, \Sigma_{i-1,d})$. This mixture pdf is considered as the proposal pdf from which samples are obtained when the weights are calculated.

Table 4
AIS with mixture Gaussian proposals.

Assume: $\chi_0 = \{x_0^{(m)}, w_0^{(m)} = \frac{1}{M}\}_{m=1}^M$, where $x_0^{(m)}, m = 1, \dots, M$ were obtained from a prior distribution.

Set: $\rho_{0,d}^{(m)} = \frac{\alpha_{-1,d} q_{0,d}(x_0^{(m)} | \mu_{-1,d}, \Sigma_{-1,d})}{\sum_{d=1}^D \alpha_{-1,d} q_{i,d}(x | \mu_{i-1,d}, \Sigma_{i-1,d})}$, $d = 1, \dots, D$, $m = 1, \dots, M$,

$$\alpha_{0,d} = \frac{1}{M} \sum_{m=1}^M \rho_{0,d}^{(m)}, \quad d = 1, \dots, D,$$

$$\mu_{0,d} = \frac{1}{M \alpha_{0,d}} \sum_{m=1}^M \rho_{0,d}^{(m)} x_0^{(m)}, \quad d = 1, \dots, D,$$

$$\Sigma_{0,d} = \frac{1}{M \alpha_{0,d}} \sum_{m=1}^M \rho_{0,d}^{(m)} (x_0^{(m)} - \mu_{0,d}) (x_0^{(m)} - \mu_{0,d})^\top \quad d = 1, \dots, D.$$

For $i = 1, \dots, I$

1. Generate new samples: $x_i^{(m)} \sim \pi_i^{(m)}(x) = \sum_{d=1}^D \alpha_{i-1,d} q_{i,d}(x | \mu_{i-1,d}, \Sigma_{i-1,d})$, $m = 1, \dots, M$, with $q_{i,d}(x | \mu_{i-1,d}, \Sigma_{i-1,d}) = \mathcal{N}(x | \mu_{i-1,d}, \Sigma_{i-1,d})$.
2. Calculate the unnormalized importance weights:

$$\bar{w}_i^{(m)} = \frac{p(x_i^{(m)})}{\sum_{d=1}^D \alpha_{i-1,d} q_{i,d}(x | \mu_{i-1,d}, \Sigma_{i-1,d})}, \quad m = 1, \dots, M.$$

3. Normalize the weights:

- (a) For adaptation of the necessary parameters for the next iteration proposal

$$w_i^{(m)} = \frac{\bar{w}_i^{(m)}}{\sum_{j=1}^M \bar{w}_i^{(j)}}, \quad m = 1, \dots, M.$$

Obtain: $\chi_i = \{x_i^{(m)}, w_i^{(m)}\}_{m=1}^M$.

- (b) For approximation of the target distribution as well as calculation of point estimates of unknowns

$$\tilde{w}_\tau^{(m)} = \frac{\bar{w}_\tau^{(m)}}{\sum_{\rho=0}^i \sum_{j=1}^M \bar{w}_\rho^{(j)}} \quad \tau = 0, \dots, i, \quad m = 1, \dots, M.$$

4. Approximate the target distribution:

$$\tilde{p}_i^M(x) = \sum_{m=1}^M \left(\tilde{w}_0^{(m)} \delta(x - x_0^{(m)}) + \tilde{w}_1^{(m)} \delta(x - x_1^{(m)}) + \dots + \tilde{w}_i^{(m)} \delta(x - x_i^{(m)}) \right).$$

5. Adapt the parameters of each of the mixands of the next iteration mixture Gaussian proposal:

$$\rho_{i,d}^{(m)} = \frac{\alpha_{i-1,d} q_{i,d}(x_i^{(m)} | \mu_{i-1,d}, \Sigma_{i-1,d})}{\sum_{d=1}^D \alpha_{i-1,d} q_{i,d}(x | \mu_{i-1,d}, \Sigma_{i-1,d})}, \quad d = 1, \dots, D, \quad m = 1, \dots, M,$$

$$\alpha_{i,d} = \sum_{m=1}^M w_i^{(m)} \rho_{i,d}^{(m)}, \quad d = 1, \dots, D,$$

$$\mu_{i,d} = \frac{1}{\alpha_{i,d}} \sum_{m=1}^M w_i^{(m)} \rho_{i,d}^{(m)} x_i^{(m)}, \quad d = 1, \dots, D,$$

$$\Sigma_{i,d} = \frac{1}{\alpha_{i,d}} \sum_{m=1}^M w_i^{(m)} \rho_{i,d}^{(m)} (x_i^{(m)} - \mu_{i,d}) (x_i^{(m)} - \mu_{i,d})^\top \quad d = 1, \dots, D.$$

The parameters of the proposal pdfs for the next iteration are updated using only the N samples generated from that particular proposal, $q_{i,d}$. Note that it is assumed that $D \leq M$ and M is a multiple of D . The weights used for the adaptation procedure, $\bar{w}_i^{(m)}$, are calculated as the ratio between the target and the proposal used for generation of the samples (i.e., considering only one mixand $q_{i,d}$ in the denominator). However, the weights used for approximation of the target distribution, $\tilde{w}_i^{(m)}$, account for the complete mixture as proposal. In the particular case $D = 1$, APIS results into a standard AIS, which only updates the means of the proposal pdfs. Table 6 summarizes the algorithm. We remark that the operation $\lfloor y \rfloor$, where y is an arbitrary argument, denotes the largest integer not greater than y .

Table 5
AMIS.

Assume: $\chi_0 = \{x_0^{(m)}, w_0^{(m)} = \frac{1}{M}\}_{m=1}^M$, where $x_0^{(m)}, m = 1, \dots, M$ were obtained from a prior distribution.

Set: $\mu_0 = \frac{1}{M} \sum_{m=1}^M x_0^{(m)}$.

$\Sigma_0 = \Sigma$, where Σ is a predefined covariance matrix of the prior.

For $i = 1, \dots, I$

1. Generate new samples: $x_i^{(m)} \sim \pi_i^{(m)}(x) = q_i(x | \mu_{i-1}, \Sigma_{i-1})$, $m = 1, \dots, M$.
2. Calculate the unnormalized importance weights:

$$\bar{w}_i^{(m)} = \frac{p(x_i^{(m)})}{\frac{1}{i+1} \sum_{j=0}^i q_j(x_j^{(m)} | \mu_{j-1}, \Sigma_{j-1})}, \quad m = 1, \dots, M,$$

and recalculate the previous ones:

$$\bar{w}_\tau^{(m)} = \frac{p(x_\tau^{(m)})}{\frac{1}{i+1} \sum_{j=0}^i q_j(x_\tau^{(m)} | \mu_{j-1}, \Sigma_{j-1})}, \quad \tau = 0, \dots, i-1, \quad m = 1, \dots, M.$$

3. Normalize the weights (for both adaptation of the necessary parameters for the next iteration proposal and approximation of the target distribution):

$$w_\tau^{(m)} = \frac{\bar{w}_\tau^{(m)}}{\sum_{j=0}^i \sum_{m=1}^M \bar{w}_\rho^{(j)}} \quad \tau = 0, \dots, i, \quad m = 1, \dots, M.$$

Obtain: $\chi_i = \{x_i^{(m)}, w_i^{(m)}\}_{m=1}^M$.

4. Approximate the target distribution:

$$\tilde{p}_i^M(x) = \sum_{m=1}^M \left(w_0^{(m)} \delta(x - x_0^{(m)}) + w_1^{(m)} \delta(x - x_1^{(m)}) + \dots + w_i^{(m)} \delta(x - x_i^{(m)}) \right).$$

5. Adapt the parameters for the next proposal:

$$\mu_i = \sum_{m=1}^M \sum_{\tau=0}^i w_\tau^{(m)} x_\tau^{(m)},$$

$$\Sigma_i = \sum_{m=1}^M \sum_{\tau=0}^i w_\tau^{(m)} (x_\tau^{(m)} - \mu_\tau) (x_\tau^{(m)} - \mu_\tau)^\top.$$

4.6. Computational cost

The computational cost of the AIS techniques is determined by the total number of samples drawn at each iteration, M , the number of proposal functions used for generation of samples, D , and the total number of iterations, I . We remark that I is also the number of adaptations of each proposal function. We define the total number of samples used for estimation of the target distribution as $M_I = M(I + 1)$, where we have taken into account the samples obtained at the initialization step of the algorithm.

In this paper, we compare the complexity of the different methods discussed in the previous section in terms of the computational cost associated to the calculation of the weights. Clearly, there are other operations which affect the computational cost such as the generation of samples, the update of parameters for the next iteration proposal, or the resampling step in the standard PMC. For the generation of samples, and given that all algorithms are assumed to use Gaussian distributions, we consider that they have the same computational complexity for this operation with the disclaimer that, in the case of the AIS with mixture Gaussian proposal pdfs, there is need for generation of a random number per sample to choose the component of the mixture to draw the sample from. The update of parameters can be assumed equivalent

Table 6
APIS.

Assume: $\chi_0 = \{x_0^{(m)}, w_0^{(m)} = \frac{1}{M}\}_{m=1}^M$, where $x_0^{(m)}, m = 1, \dots, M$ were obtained from a prior distribution.

Set: $\mu_{0,d} = \frac{1}{M} \sum_{n=1}^N x_0^{(d-1)N+n}, d = 1, \dots, D$.

$\Sigma_{0,d} = \Sigma_d, d = 1, \dots, D$, where Σ_d is a predefined covariance matrix of the prior.

For $i = 1, \dots, I$

1. Generate new samples: $x_i^{(m)} \sim \pi_i^{(m)}(x) = q_{i,d}(x | \mu_{i-1,d}, \Sigma_{i-1,d})$,
 $d = \lfloor \frac{m-1}{N} \rfloor + 1, m = 1, \dots, M$.

2. Calculate the unnormalized importance weights:

- (a) For adaptation of the necessary parameters for the next iteration proposal

$$\tilde{w}_i^{(m)} = \frac{p(x_i^{(m)})}{q_{i,d}(x_i^{(m)} | \mu_{i-1,d}, \Sigma_{i-1,d})}, \quad d = \lfloor \frac{m-1}{N} \rfloor + 1,$$

$$m = 1, \dots, M.$$

- (b) For approximation of the target distribution as well as calculation of point estimates of unknowns

$$\hat{w}_i^{(m)} = \frac{p(x_i^{(m)})}{\frac{1}{D} \sum_{d=1}^D q_{i,d}(x_i^{(m)} | \mu_{i-1,d}, \Sigma_{i-1,d})}, \quad m = 1, \dots, M.$$

3. Normalize the weights:

- (a) For adaptation of the necessary parameters for the next iteration proposal

$$w_i^{(m)} = \frac{\tilde{w}_i^{(m)}}{\sum_{j=d+1}^{d+N} \tilde{w}_i^{(j)}}, \quad d = \lfloor \frac{m-1}{N} \rfloor N, \quad m = 1, \dots, M.$$

$$\text{Obtain: } \chi_i = \{x_i^{(m)}, w_i^{(m)}\}_{m=1}^M.$$

- (b) For approximation of the target distribution as well as calculation of point estimates of unknowns

$$\hat{w}_\tau^{(m)} = \frac{\tilde{w}_\tau^{(m)}}{\sum_{\rho=0}^i \sum_{j=1}^M \tilde{w}_\rho^{(j)}}, \quad \tau = 0, \dots, i, \quad m = 1, \dots, M.$$

4. Approximate the target distribution:

$$\tilde{p}_i^M(x) = \sum_{m=1}^M \left(\tilde{w}_0^{(m)} \delta(x - x_0^{(m)}) + \tilde{w}_1^{(m)} \delta(x - x_1^{(m)}) + \dots \right. \\ \left. + \tilde{w}_i^{(m)} \delta(x - x_i^{(m)}) \right).$$

5. Adapt the parameters for the next proposal:

$$\mu_{i,d} = \sum_{n=1}^N w_i^{(d-1)N+n} x_i^{(d-1)N+n}, \quad d = 1, \dots, D,$$

$$\Sigma_{i,d} = \Sigma_{i-1,d} \quad d = 1, \dots, D.$$

in complexity for all methods and is more critical for large values of M , D and I . Here, for simplicity, we do not consider them. The same holds for the resampling step of the PMC, since it will depend on the specific type of resampling and will be ignored to make the comparison more straightforward.

Table 7 shows the number of operations needed for calculation of the weights for each of the considered methods. This calculation corresponds to obtaining the ratio of target distribution to the proposal distribution. It is determined at each iteration and for each generated sample (we note that AMIS also recomputes at each iteration the weights corresponding to the previous samples). Therefore, the table reflects the total number of evaluations of the target and proposal pdfs for the different techniques and the overall complexity is the result of the addition of both numbers. For all the

methods, the number of target evaluations is $M_I = M(I + 1)$, i.e., for each sample at each iteration we evaluate the target once. The evaluation of the denominator varies with the considered method. The standard PMC appears as the *lightest* algorithm and the number of evaluations of the proposal linearly grows with D (or M , which coincide in this case) and I . In this case, there is one proposal per sample that is evaluated at each iteration. In the AIS with mixture Gaussian proposals the evaluation of the denominator for a particular sample at a given iteration takes into account the D components of the mixture and therefore the number of proposal functions that must be evaluated at each step is proportional to D . The same argument holds for APIS, where the calculation of the weights reflects the appropriateness of the sample to the mixture of D components. Finally, for the AMIS, the computational load increases per iteration, since the evaluation of the denominator takes into account the adequacy of the samples to all the previous proposal pdfs and the current one. In addition, one also recalculates all the previous weights. This latter operation involves evaluation of the new proposal with respect to all old samples (one assumes that evaluations of previous proposals are stored and no recalculation is needed). Therefore, in AMIS, it is computationally expensive to adapt several times the proposal function, i.e., large values of I increase its complexity.

5. Examples

In this section we provide computer simulation results corresponding to three different problems. Namely, we first discuss the well known “banana-shaped” distribution problem [17] as proof of concept. Then, we consider a high-dimensional setting with a multimodal target distribution, and finally we discuss an example of an AR model in a non-Gaussian noise scenario.

5.1. Banana-shaped distribution

We consider as target distribution the bi-dimensional “banana-shaped” distribution [17], which is a benchmark function in the literature due to its nonlinear nature. Mathematically, it is expressed as

$$p(x_1, x_2) \propto \exp\left(-\frac{1}{2\eta_1^2} (4 - Bx_1 - x_2^2)^2 - \frac{x_1^2}{2\eta_2^2} - \frac{x_2^2}{2\eta_3^2}\right), \quad (14)$$

where, in this case, we set $B = 10$, $\eta_1 = 4$, $\eta_2 = 5$ and $\eta_3 = 5$. The objective is to estimate the expected value $E[X]$, where $X = [X_1, X_2] \sim p(x_1, x_2)$, by applying different Monte Carlo approximations. We can approximately obtain the true value $E[X] \approx [-0.4845, 0]^T$ using an exhaustive deterministic numerical method (with an extremely thin grid), in order to obtain the mean square error (MSE) for comparison of the standard PMC, the AIS with mixture Gaussian proposal pdfs, the AMIS and the APIS (labeled in the figures and referred in this section to as PMC, Mix-AIS, AMIS, and APIS, respectively).

For simplicity, we consider Gaussian proposal distributions for all the algorithms. The initialization is performed by randomly drawing the parameters of the Gaussians, with the mean of the j -th prior given by $\mu_{-1,j} \sim \mathcal{U}([-6, 3] \times [-4, 4])$ and its covariance matrix given by $\Sigma_{-1,j} = [\sigma_{j,1}^2 \quad 0; 0 \quad \sigma_{j,2}^2]^T$. We note that the subscript -1 indicates that the distribution is a prior and the subscript j denotes the mixture in the initial mixture (i.e., in standard PMC we have M priors, one per sample; in Mix-AIS we have D components of the mixture; in AMIS we have one proposal; and in APIS we have D proposal pdfs). We contemplate two cases: an *isotropic* setting where $\sigma_{j,1} = \sigma_{j,2} = \sigma \in \{1, 2, \dots, 10\}$, and an *anisotropic* case with random selection of the parameters to test the robustness of the methods and where $\sigma_{j,1} \sim \mathcal{U}([1, 20])$ and

Table 7Computational cost: Total number of target and proposal evaluations when drawing M samples per iteration and after I iterations.

Method	M	D	Target evaluations	Proposal evaluations
PMC	$M = D$	$D \geq 2$	$M(I + 1) = D(I + 1) = M_I$	$M(I + 1) = D(I + 1) = M_I$
AIS with mixture Gaussians	M	$D \geq 2$	$M(I + 1) = M_I$	$MD(I + 1) = M_I D$
AMIS	M	$D = 1$	$M(I + 1) = M_I$	$M \sum_{i=0}^M (2i + 1) = M(I + 1)^2 = M_I(I + 1)$
APIS	$M = ND$	$D \leq \frac{M}{2}$	$M(I + 1) = ND(I + 1) = M_I$	$MD(I + 1) = ND^2(I + 1) = M_I D$

Table 8Bi-dimensional banana-shaped distribution example: Best and worst results in terms of MSE, obtained with the different techniques for different values of σ (**Test 1**). The smallest MSE for each σ is boldfaced.

Algorithm		$\sigma = 0.5$	$\sigma = 1$	$\sigma = 2$	$\sigma = 3$	$\sigma = 5$	$\sigma = 10$	$\sigma = 70$	$\sigma_{i,j} \sim \mathcal{U}([1, 20])$
PMC	Worst	0.0670	0.0461	0.0209	0.0093	0.0055	0.0072	9.4749	0.1065
	Best	0.0210	0.0164	0.0069	0.0016	0.0015	0.0011	0.0262	0.0026
Mix-AIS	Worst	3.5772	0.0113	0.0044	0.0066	0.0174	0.0267	0.0913	0.0103
	Best	0.0092	0.0020	0.0018	0.0035	0.0034	0.0055	0.0138	0.0025
AMIS	Worst	0.0040	0.0039	0.0040	0.0016	0.0011	0.0012	0.0035	0.0013
	Best	0.0023	0.0028	0.0023	0.0009	0.0003	0.0004	0.0023	0.0007
APIS	Worst	0.0443	0.0437	0.0013	0.0007	0.0150	0.2486	13.7161	0.0762
	Best	0.0051	0.0001	0.0002	0.0001	0.0002	0.0006	0.0361	0.0004

$\sigma_{j,2} \sim \mathcal{U}([1, 20])$. Recall that the standard PMC and the APIS do not adapt the covariance matrices along the iterations but consider them fixed.

For each of the algorithms, at each iteration, M samples are generated. We run three different experiments:

Test 1: Fixed total number of samples M_I : We keep fixed the total number of samples $M_I = M(I + 1) = 2 \cdot 10^5$ (namely, the total number of generated samples and evaluation of the target) and vary the value of σ . Different combinations of parameters are examined:

- For PMC: $M \in \{50, 100, 10^3, 5 \cdot 10^3\}$ and $I = \frac{2 \cdot 10^5}{M}$. Note that $D = M$ in the basic PMC.
- For Mix-AIS: $M \in \{100, 200, 10^3, 2 \cdot 10^3, 5 \cdot 10^3\}$, $I = \frac{2 \cdot 10^5}{M}$ and $D \in \{10, 50, 100\}$.
- For AMIS: $M \in \{500, 10^3, 2 \cdot 10^3, 5 \cdot 10^3, 10^4\}$ and $I = \frac{2 \cdot 10^5}{M}$. Recall that $D = 1$ in AMIS.
- For APIS: $M \in \{100, 200, 10^3, 2 \cdot 10^3, 5 \cdot 10^3\}$, $I = \frac{2 \cdot 10^5}{M}$ and $D \in \{50, 100\}$.

The parameter N for the APIS method is set such that $N = \frac{M}{D} \geq 2$. The range of values of the parameters are chosen, after a preliminary study, in order to obtain the best performance from each technique.

Test 2: Fixed M and σ , varying I : We set $M = 10^3$ and $\sigma = 5$. The total number of samples $M_I = M(I + 1)$ is changed depending on the value of I , which represents the total number of adaptation steps of the algorithms.

Test 3: Fixed I and σ , varying M : We set $I = 40$ and $\sigma = 5$. The total number of samples $M_I = M(I + 1)$ is changed depending on the value of M , which represents the number of samples generated at each iteration.

For **Test 2** and **Test 3**, we set $D = 100$ (the number of proposals, or, what is the same, components in the proposal) for the Mix-AIS and the APIS schemes. We recall that, in Mix-AIS and AMIS, the covariance matrices are also updated during the adaptation, and therefore the value σ for these methods only represents the initial parameters.

The results are averaged over 500 independent simulations, for each combination of parameters. Table 8 shows the smallest and highest MSE values obtained in the estimation of the expected value of the target, averaged between the two components of $E[X]$, achieved by the different methods in the first experiment **Test 1**. The smallest MSE in each column (each σ) is boldfaced. The

log-MSEs as a function of σ are shown in Figs. 3(a)–(b), while Figs. 3(c)–(d) show the log-MSE as function of I (**Test 2**) and of M (**Test 3**), respectively. Finally, Fig. 4 displays the final configurations of the means for the different algorithms. Note that for the Mix-AIS and AMIS the covariance matrices are also adapted and the circles show approximately 85% of probability mass. In this example, AMIS and APIS provide the best results. AMIS works better starting with larger initial parameters (since AMIS adapts the variances as well), whereas APIS prefers smaller initial parameters. In general, APIS needs to tune the parameter $N = \frac{M}{D}$ as function of the variance σ . For instance, with $\sigma = 70$ the results show that a larger N is necessary. However, due to this additional degree of freedom compared to the standard PMC, APIS is able to reach good performance.

5.2. Multimodal target distribution

In this section, we consider a high-dimensional multimodal distribution, $x \in \mathbb{R}^{10}$, defined as mixture of three multivariate Gaussians

$$p(x) \propto \frac{1}{3} \sum_{n=1}^3 \mathcal{N}(x | v_n, \Lambda_n), \quad (15)$$

with $v_n = [v_{n,1}, \dots, v_{n,10}]^\top$ representing the 10-dimensional vector of means corresponding to the n -th component in the multivariate distribution and $\Lambda_n = \sigma_n \mathbb{I}_{10}$, $n = 1, 2, 3$, denoting the corresponding 10-dimensional vector of covariance matrices, where \mathbb{I}_{10} is the 10×10 identity matrix. We set $v_{1,j} = 6$, $v_{2,j} = -5$ with $j = 1, \dots, 10$, and $v_3 = [1, 2, 3, 4, 5, 5, 4, 3, 2, 1]^\top$. Moreover, we set $\sigma_n = 3$, for all $n = 1, 2, 3$. For this target distribution $p(x)$, one can analytically calculate the expected value, which is given

$$\text{by } E[X] = \left[\frac{2}{3}, 1, \frac{4}{3}, \frac{5}{3}, 2, 2, \frac{5}{3}, \frac{4}{3}, 1, \frac{2}{3} \right]^\top, \text{ where } X \sim p(x).$$

As in the previous example, we consider Gaussian proposal functions for all the compared methods. The initialization is performed by randomly drawing the means of the Gaussians, with the mean of the j -th pdf given by $\mu_{-1,d} \sim \mathcal{U}([-10 \times 10]^{10})$, $d = 1, \dots, D$. We note that, unlike the previous example, this is a good initialization since the hyper-rectangle $[-10 \times 10]^{10}$ contains all the modes of the target distribution. We also use different initial covariance matrices, with the one corresponding to the d -th proposal pdf being, $\Sigma_{-1,d} = \sigma \mathbb{I}_{10}$ and $\sigma \sim \mathcal{U}([1, 10])$.

We test different combinations of parameters keeping the total number of used samples fixed, $M_I = M(I + 1) = 4 \cdot 10^5$. We

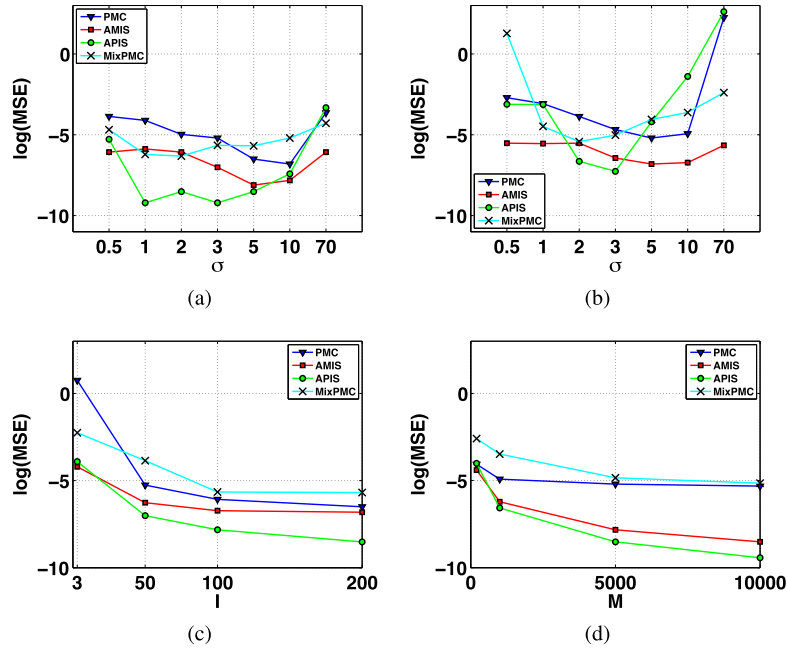


Fig. 3. Bi-dimensional banana-shaped distribution example: Log-MSE of the different algorithms (triangles: PMC; x-mark: Mix-AIS; squares: AMIS; circles: APIS): (a) Best results for Test 1; (b) Worst results for Test 1 (as a function of σ); (c) Test 2 (as a function of I); (d) Test 3 (as a function of M).

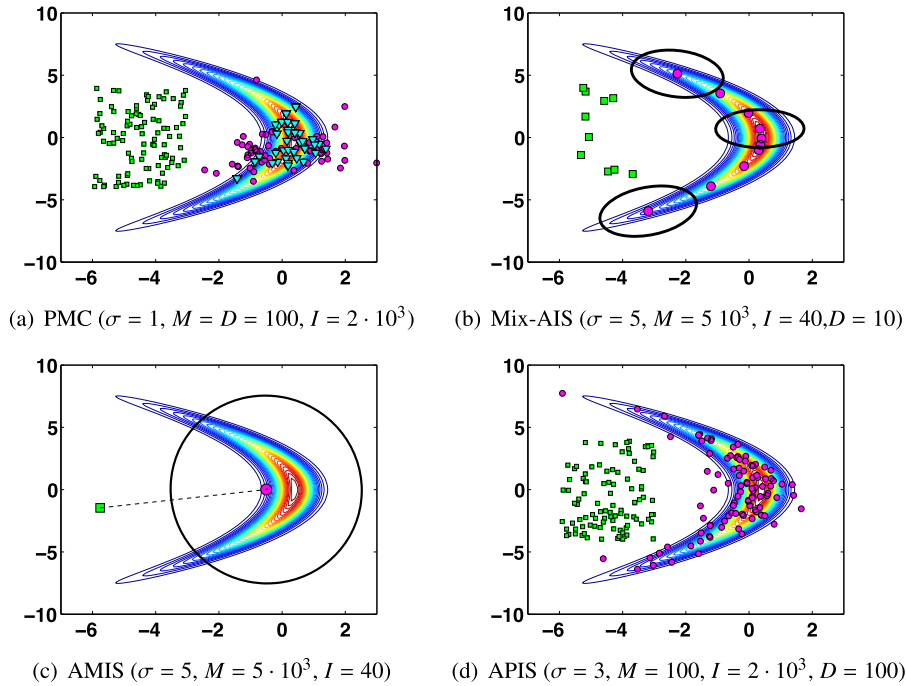


Fig. 4. Bi-dimensional banana-shaped distribution example: Final configurations of the location parameters (means) of the proposal distributions. The Mix-AIS and AMIS techniques also adapt the covariance matrices (the circles show approximately 85% of the probability mass).

evaluate different values of M , $M \in \{100, 200, 500, 10^3, 2 \cdot 10^3, 5 \cdot 10^3, 10^4, 2 \cdot 10^4, 4 \cdot 10^4, 10^5\}$, and as a consequence, $I = \frac{4 \cdot 10^5}{M}$.

We recall that $D = M$ in PMC and $D = 1$ in AMIS. For the Mix-AIS and APIS methods we use $D \in \{10, 100, 200\}$. The parameter N for APIS is set to $N = \lfloor \frac{M}{D} \rfloor \geq 2$. We run 500 independent simulations and compute the MSE of $E[X]$ (we average the MSEs of each component). The worst, the best and the averaged results in terms of MSE are shown in Table 9. As seen, for a multimodal target if a good initialization is used, the standard PMC obtains better performance than in the previous example, providing the minimum MSE. APIS also shows good performance achieving the best averaged re-

sults. APIS gets its worst result in the extreme case $N = \frac{M}{D} = 2$ (smallest possible value). AMIS suffers from the multimodal scenario since it often gets trapped in a specific mode.

Finally, Fig. 5 depicts a particular bi-dimensional slice of the target pdf (with $x = [x_1, \dots, x_{10}]^T \in \mathbb{R}^{10}$), together with the last configurations of the location parameters (means) of the proposal pdfs for PMC and APIS in one specific run. The slice is a function of x_1 and x_{10} , and it is obtained keeping fixed the other variables as $x_{2:5} = -9.85$, $x_{6:9} = 8.5$. Moreover, in these plots, we set $D = 200$ for both, $\mu_{-1,d} \sim \mathcal{U}([-6 \times 6]^{10})$ (initial means, $d = 1, \dots, D$), $\sigma \sim \mathcal{U}([1, 6])$, and $I = \frac{2 \cdot 10^5}{M}$ (in APIS, $M = ND$). It is clear that the PMC

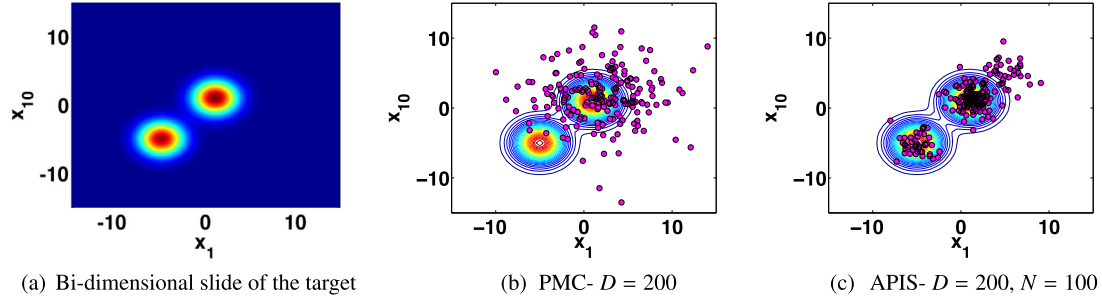


Fig. 5. Multimodal target distribution example ($x \in \mathbb{R}^{10}$): (a) Bi-dimensional slice of the multimodal target pdf as a function of x_1 and x_{10} while fixing $x_{2:5} = -9.85$ and $x_{6:9} = 8.5$; (b) Final configuration of the location parameters (means) of the proposal pdfs for PMC; and (c) Final configuration of the location parameters (means) of the proposal pdfs for APIS.

Table 9

Multimodal target distribution example ($x \in \mathbb{R}^{10}$): The worst, best and averaged results in terms of MSE for the different techniques and after testing several combinations of parameters. The smallest MSE for each row is boldfaced.

Results	PMC	Mix-AIS	AMIS	APIS
Worst	16.05	16.97	21.07	32.51
Best	0.0858	7.68	13.70	0.3857
Average	7.52	10.92	16.92	2.82

fails in detecting the two modes while APIS is able to lock both of them (recall that is a slice of a pdf in \mathbb{R}^{10}).

5.3. AR filters with non-Gaussian noise

In this example, we consider the use of an autoregressive (AR) model contaminated by a non-Gaussian noise. This type of filters are often used for modeling financial time series, where the noise is assumed to follow the so-called *generalized hyperbolic distribution* [63,64]. Namely, we consider the AR filter

$$y_k = \sum_{r=1}^R x_r y_{k-r} + u_k, \quad k = 1, \dots, K, \quad (16)$$

where u_k represents a heavy-tailed driving noise following a generalized hyperbolic distribution

$$u_k \sim p(u) \propto e^{\beta(u-\mu)} \frac{B_{\lambda-\frac{1}{2}}(\alpha\sqrt{\delta^2+(u-\mu)^2})}{(\sqrt{\delta^2+(u-\mu)^2})^{\frac{1}{2}-\lambda}}, \quad (17)$$

with B_λ denoting the modified Bessel function of the second kind [65]. We set $\alpha = 2$, $\beta = 1$, $\lambda = 0.5$, $\mu = -1$, and $\delta = 1$. In the next two subsections, we study two different cases separately, corresponding to $R = 4$ and $R = 30$, respectively.

5.3.1. Case $R = 4$

We synthetically generate $K = 200$ observations, $\{y_k\}_{k=1}^K$,¹ setting $x = [x_1, x_2, x_3, x_4]^\top = [0.5, 0.1, -0.8, 0.1]^\top$ and $y_r = 0$ for $r = 1, 2, 3, 4$. Assuming improper uniform priors over the coefficient, the objective is to approximate the posterior of vector $x = [x_1, x_2, x_3, x_4]^\top$. We note that with $K = 200$, the target posterior is quite sharp and concentrated around the real values x_i , $i = 1, 2, 3, 4$, thus we assume the vector $x = [0.5, 0.1, -0.8, 0.1]^\top$ as the true expected value of the posterior.

¹ For simulation of i.i.d. samples of the generalized hyperbolic noise, we have applied a fast and efficient MCMC technique (the FUSS algorithm [66]) drawing samples from univariate distributions. After few iterations, the resulting samples are roughly independent.

Table 10

AR model with non-Gaussian noise example with $R = 4$ unknown coefficients (i.e., $x \in \mathbb{R}^4$): The worst, best and averaged results in terms of MSE for the different techniques and after testing several combinations of parameters. The smallest MSE for each row is boldfaced.

Results	PMC	Mix-AIS	AMIS	APIS	MH
Worst	10.57	20.23	0.7141	0.0702	0.3021
Best	0.0235	0.1266	0.3492	0.0539	
Average	2.054	5.207	0.4301	0.0683	-

As in the previous examples, the methods use Gaussian proposal pdfs and the initial samples are selected randomly. In particular, for the j -th pdf its mean is $\mu_{-1,j} \sim \mathcal{U}(\mathcal{R})$, where $\mathcal{R} = [-5, 5] \times [-5, 5] \times [-5, 5] \times [-5, 5]$, and its covariance matrix $\Sigma_{-1,j} = \xi_j^2 \mathbb{I}_4$, with $\xi_j^2 = [\sigma_{j,1}^2, \sigma_{j,2}^2, \sigma_{j,3}^2, \sigma_{j,4}^2]^\top$ and $\sigma_{j,r} \sim \mathcal{U}([1, 6])$ for $r = 1, \dots, 4$.

We test different combinations of parameters keeping the total number of used samples fixed, $M_I = 2 \cdot 10^6$. We evaluate different values of M , $M \in \{200, 2 \cdot 10^3, 5 \cdot 10^3, 10^4, 2 \cdot 10^4, 4 \cdot 10^4, 10^5\}$, and as a consequence, $I = \frac{2 \cdot 10^6}{M}$.

We recall that $D = M$ in PMC and $D = 1$ in AMIS. For the Mix-AIS and APIS methods we use $D \in \{10, 200, 500\}$. The parameter N for APIS is $N = \lfloor \frac{M}{D} \rfloor \geq 2$. We also run the Metropolis–Hastings (MH) algorithm [2,16] with a random-walk Gaussian proposal distribution using the same initialization parameters as for the rest of methods. We consider a number of iterations for MH such that its execution time is comparable to the other techniques (one can find that $I_{MH} \approx 4 \cdot 10^4$).²

We run 500 independent simulations and compute the MSE in the estimation of x (we average the MSEs of each component). The worst, best and averaged results in terms of MSE, obtained by the different methods, are shown in Table 10. The best result is provided by the standard PMC since the initialization of the samples is in a region that contains the mode of the posterior distribution, which is critical in the performance of the method. APIS provides the smallest difference between the worst and best results.

Fig. 6 illustrates the histograms of the estimations of the expected value of the target pdf (namely, the outputs of the algorithms in this example) obtained with the different techniques and some specific values of the parameters indicated in the captions of the subplots. The true values $x = [0.5, 0.1, -0.8, 0.1]^\top$ are depicted with vertical dashed lines. This figure helps to understand that the results in Table 10 of AMIS do not properly reflect the overall performance of this method. Namely, as shown in Fig. 6(b), AMIS often provides values close to the true ones but in some particular runs, depending on the initial parameters, the method does

² The MH algorithm is a sequential method compared to the AIS methods which are more parallelizable.

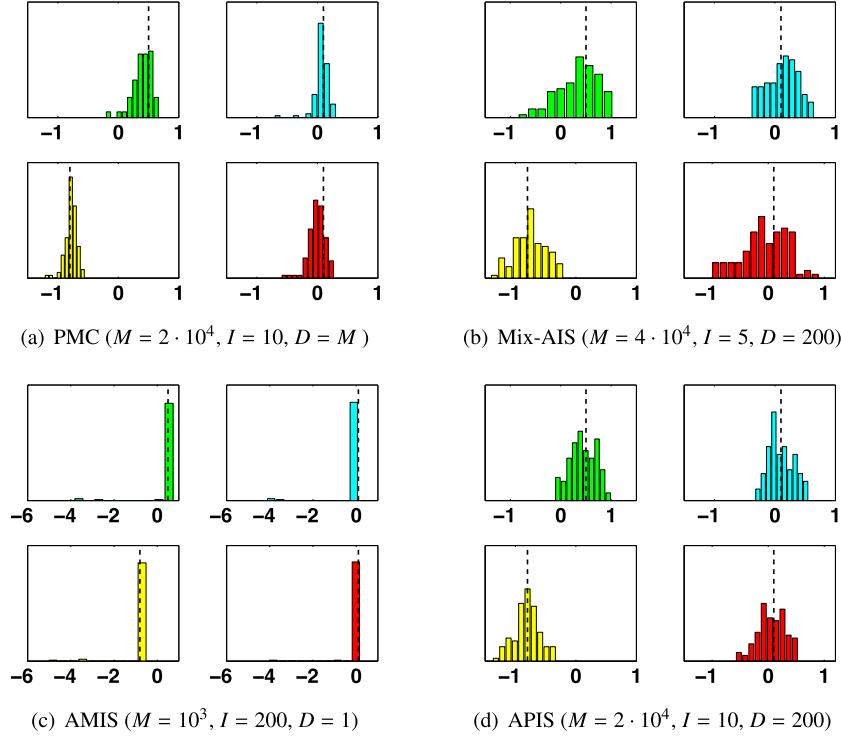


Fig. 6. AR model with non-Gaussian noise example with $R = 4$ unknown coefficients (i.e., $x \in \mathbb{R}^4$): Histograms of the estimations of the expected value of the target pdf obtained with the different techniques. The true values of the components of x are shown with vertical dashed lines. The used combination of parameters is indicated in each subplot caption.

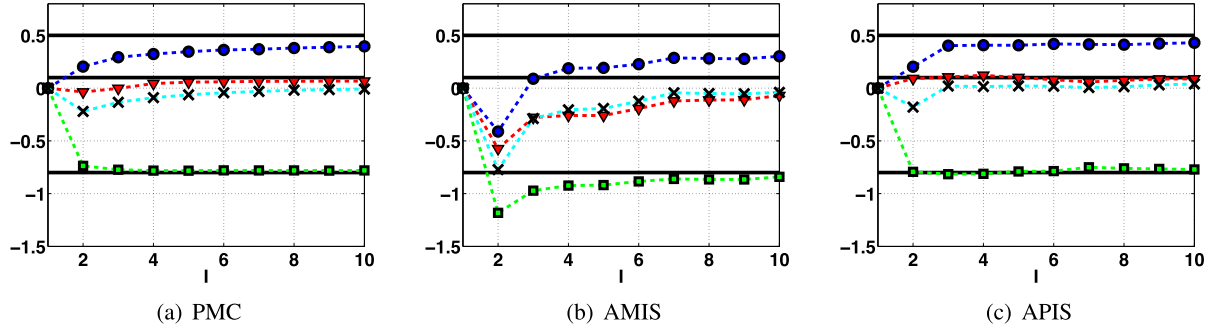


Fig. 7. AR model with non-Gaussian noise example with $R = 4$ unknown coefficients (i.e., $x \in \mathbb{R}^4$): Averaged curves of convergence (in terms of estimation of the x values) of PMC, AMIS and APIS, respectively, as function of the number of iteration I . The solid lines show the true values of the components of $x = [0.5, 0.1, -0.8, 0.1]^T$.

not converge and the results deteriorate the MSE performance. However, as seen in Fig. 6(b), AMIS may provide excellent performance. Finally, Fig. 7 displays the averaged convergence curves (in terms of estimation of the x values) of PMC, AMIS and APIS, respectively.

5.3.2. Case $R = 30$

We consider the case $x \in \mathbb{R}^{30}$, i.e., $R = 30$, with $K = 50$ observations. The data are generated setting $x_r = 1.7e^{-r}$, and $y_r = 1$, with $r = 1, \dots, 30$. These coefficients ensure that the AR filter is stable. We use Gaussian proposal densities with initial samples selected randomly. In particular, for the j -th prior its mean is $\mu_{-1,j} \sim \mathcal{U}(\mathcal{R})$, where

$$\mathcal{R} = [-2, 4]^{30} = [-2, 4] \underbrace{\times \dots \times}_{30} [-2, 4], \tag{18}$$

and its covariance matrix $\Sigma_{-1,j} = \xi_j^2 \mathbb{I}_{30}$, with $\xi_j^2 = [\sigma_{j,1:30}^2]^\top$ and $\sigma_{j,r} \sim \mathcal{U}([0, 1])$ for $r = 1, \dots, 30$. We test different combinations of

parameters keeping the total number of used samples fixed, $M_I = 2 \cdot 10^5$:

- For PMC: $M \in \{10^3, 5 \cdot 10^3, 10^4, 2 \cdot 10^4, 4 \cdot 10^4, 5 \cdot 10^4\}$ (recall $M = D$).
- For Mix-AIS: $D \in \{10, 200, 10^3\}$ and $M \in \{100, 500, 10^3\}$.
- For AMIS: $M \in \{10^3, 5 \cdot 10^3, 10^4, 2 \cdot 10^4, 4 \cdot 10^4, 5 \cdot 10^4\}$ (recall $D = 1$).
- For APIS: $D \in \{200, 10^3\}$ and $N \in \{2, 10, 100, 200, 500\}$ (recall $M = ND$).

The total number of iterations is set to $I = \frac{M_I}{M} = \frac{2 \cdot 10^5}{M}$. We run 500 independent simulations and compute the MSE in the estimation of x (we average the MSEs of each component). The worst, best and averaged results in terms of MSE are shown in Table 11. In this case, the performance of PMC, AMIS and APIS are quite similar, but APIS seems to be more robust. We remark that in this example we test APIS even with small values of $N \in \{2, 10\}$ ($N = 2$ is the smallest possible value) which appears completely inappropriate for the dimension of the problem ($x \in \mathbb{R}^{30}$). How-

Table 11

AR model with non-Gaussian noise example with $R = 30$ unknown coefficients (i.e., $x \in \mathbb{R}^{30}$): The worst, best and averaged results in terms of MSE for the different techniques and after testing several combinations of parameters. The smallest MSE for each row is boldfaced.

Results	PMC	Mix-AIS	AMIS	APIS
Worst	22.7401	15.034	13.670	6.371
Best	1.394	3.805	1.418	0.934
Average	6.214	7.221	4.256	2.221

ever, the used scale parameters in this example are quite small, $\sigma_{j,r} \sim \mathcal{U}([0, 1])$, explaining why APIS obtains satisfactory results (compared to the other techniques) even with small N . Moreover, in this numerical example we can argue that APIS takes advantage of the deterministic mixture approach in a more efficient way than AMIS, considering in terms of computational effort and capability of reaching different modes.

5.4. Discussion

In this section, we provide some considerations about the simulation results and, more in general, the different techniques. The standard PMC method is less costly and the easiest algorithm to implement. Its performance is sensitive to the initialization, i.e., a *good* initialization in a finite space containing the modes of the target distribution is critical for its adequate performance (see for instance, Sections 5.2 and 5.3). If the mass of probability is concentrated in a small area in the domain of the target distribution, PMC performs better with a greater D ($D = M$). Moreover, with $D \rightarrow \infty$, the resampling procedure provides virtually i.i.d. samples from the target distribution. However, in the first example in Section 5.1 the results are worse than the other considered techniques. This is probably the result of a *bad* initialization, where samples were distributed uniformly on a rectangle which did not contain high probability regions.

AMIS seems to be stable and robust in terms of convergence and results, although its computational cost grows quickly with I . For this reason and for ease in the adaptation of parameters, when we keep constant $M_I = MI$, AMIS works better with a greater value M compared to APIS and PMC. Moreover, since AMIS adapts the covariance matrix as well, larger values for the initial variances are more preferable than in APIS and PMC. Keeping constant $M_I = MI$, there exists a relationship between the value of M and the values for the initial variances: larger initial parameters need of a larger value of M , while smaller initial variances need smaller M (and larger I). AMIS seems to be able to lock easily tight sharp modes when compared to the other algorithms. However, APIS and PMC seem more appropriate to handle multimodal target distributions, since they use D mixands in the proposal pdfs that allow for better exploration of state of the system (in AMIS, we have $D = 1$).

Furthermore, in these numerical examples, the Mix-AIS displays worse results than other techniques. The adaptation procedure of the proposal seems more difficult, slower and quite sensitive to the initial conditions. Unlike PMC and APIS, the proposal adaptation seems to work better with smaller values of D . Moreover, since Mix-AIS also updates the weights of the mixture, it seems to suffer a loss of diversity in the population, similar to that of the PMC scheme. As in AMIS, for a good adaptation one usually needs larger values of M as opposed to PMC and APIS.

Finally, APIS delivers in general good results especially with multimodal target distributions. APIS shares with PMC several properties, however it seems less dependent on the initial conditions. Its performance clearly depends on the value of N (such that $M = ND$). If N is too small, the adapted proposal pdfs become closer to random walks and the results deteriorate. With a large N , APIS becomes closer to a standard AIS. However, for

given a number D of proposal distributions with chosen parameters, there exists an optimal value N (keeping constant the number of total samples $M_I = MI = NDI$) [25]. For larger variances, a larger N is needed, whereas with smaller variances a smaller value of N provides better results.

6. Conclusions

This work provides a comprehensive review of the basics of adaptive importance sampling (AIS) in the field of signal processing. An extensive survey of its state-of-the-art is provided, the general problem formulation and procedure is revisited. Moreover, we have studied in depth four specific methodologies that have obtained particular interest in the literature and have been applied successfully in signal processing. Several connections and differences have been highlighted. We have compared these different AIS schemes in three relevant numerical examples, providing insights of the advantages and disadvantages. These considerations together with the general outline discussed in the first part of this work, represent a useful tool for designing novel AIS techniques that may share features and overcome drawbacks of existing methodologies.

Acknowledgments

M.F.B. would like to thank the support of the National Science Foundation under Award CCF-0953316. L.M. and J.C. were funded by ERC grant 239784 and AoF grant 251170.

References

- [1] J.J.K.O. Ruanaidh, W.J. Fitzgerald, Numerical Bayesian Methods Applied to Signal Processing, Springer, New York, 2012.
- [2] C.P. Robert, G. Casella, Monte Carlo Statistical Methods, Springer, New York, 1999.
- [3] L. Martino, J. Míguez, Generalized rejection sampling schemes and applications in signal processing, Signal Process. 90 (11) (2010) 2981–2995.
- [4] W.R. Gilks, S. Richardson, D.J. Spiegelhalter (Eds.), Markov Chain Monte Carlo in Practice, Chapman & Hall, New York, 1996.
- [5] F. Liang, C. Liu, R. Caroll, Advanced Markov Chain Monte Carlo Methods: Learning from Past Samples, Wiley Series in Computational Statistics, Wiley, England, 2010.
- [6] W.J. Fitzgerald, Markov chain Monte Carlo methods with applications to signal processing, Signal Process. 8 (1) (2001) 3–18.
- [7] A. Doucet, N. de Freitas, N. Gordon (Eds.), Sequential Monte Carlo Methods in Practice, Springer, New York, 2001.
- [8] B. Chen, W.B. Heinzelman, M. Liu, A.T. Campbell, Editorial of special issue on wireless sensor networks, EURASIP J. Wirel. Commun. Netw. 2005 (4) (2005) 459–461.
- [9] G. Casella, C.P. Robert, Post-processing accept-reject samples: recycling and rescaling, J. Comput. Graph. Stat. 7 (2) (1998) 139–157.
- [10] J.S. Liu, R. Chen, Sequential Monte Carlo methods for dynamic systems, J. Am. Stat. Assoc. 93 (443) (1998) 1032–1044.
- [11] J.S. Liu, Metropolisized independent sampling with comparisons to rejection sampling and importance sampling, Stat. Comput. 6 (2) (1996) 113–119.
- [12] G. Douc, C. Robert, A vanilla Rao-Blackwellization of Metropolis-Hastings algorithms, Ann. Stat. 39 (1) (2011) 261–277.
- [13] J.S. Liu, Monte Carlo Strategies in Scientific Computing, Springer, New York, 2001.
- [14] J.M. Cornuet, J.M. Marin, A. Mira, C.P. Robert, Adaptive multiple importance sampling, Scand. J. Stat. 39 (4) (2012) 798–812.
- [15] O. Cappé, A. Guillin, C.P. Robert, Population Monte Carlo, J. Comput. Graph. Stat. 13 (2004) 907–929.
- [16] J.S. Liu, Monte Carlo Strategies in Scientific Computing, Springer, 2004.
- [17] H. Haario, E. Saksman, J. Tamminen, An adaptive Metropolis algorithm, Bernoulli 7 (2) (2001) 223–242.
- [18] L. Martino, J. Read, D. Luengo, Independent doubly adaptive rejection Metropolis sampling, in: IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP), 2014, pp. 7998–8002.
- [19] L. Holden, R. Hauge, M. Holden, Adaptive independent Metropolis-Hastings, Ann. Appl. Probab. 19 (1) (2009) 395–413.
- [20] B.D. Ripley, Stochastic Simulation, John Wiley & Sons, New York, 1987.

- [21] D.B. Rubin, A noniterative sampling/importance resampling alternative to the data augmentation algorithm for creation a few imputations when fractions of missing information are modest: the SIR algorithm, *J. Am. Stat. Assoc.* 82 (1987) 543–546.
- [22] O. Cappé, R. Douc, A. Guillin, J.-M. Marin, C.P. Robert, Adaptive importance sampling in general mixture classes, *Stat. Comput.* 18 (2008) 447–459.
- [23] J.M. Marin, P. Pudlo, M. Sedki, Consistency of the adaptive multiple importance sampling, arXiv:1211.2548, 2012, pp. 1–25.
- [24] L. Martino, V. Elvira, D. Luengo, J. Corander, An adaptive population importance sampler: learning from uncertainty, arXiv:1405.0280, 2012, pp. 1–19.
- [25] L. Martino, V. Elvira, D. Luengo, J. Corander, An adaptive population importance sampler, in: *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2014, pp. 8088–8092.
- [26] L. Martino, V. Elvira, D. Luengo, J. Corander, An adaptive population importance sampler: learning from uncertainty, *IEE Proc. F* 140 (1993) 107–113.
- [27] S. Brooks, A. Gelman, G.L. Jones, X.-L. Meng (Eds.), *Handbook of Markov Chain Monte Carlo*, CRC Press, 2011.
- [28] F. Liang, C. Liu, R. Carroll, *Advanced Markov Chain Monte Carlo Methods: Learning from Past Samples*, Wiley, 2010.
- [29] C.P. Robert, G. Casella, *Monte Carlo Statistical Methods*, Springer, 2004.
- [30] N.J. Gordon, D.J. Salmond, A.F.M. Smith, Novel approach to nonlinear/non-Gaussian Bayesian state estimation, *IEE Proc. F* 140 (1993) 107–113.
- [31] O. Cappé, S.J. Godsill, E. Moulines, An overview of existing methods and recent advances in sequential Monte Carlo, *Proc. IEEE* 95 (2007) 899–924.
- [32] P.M. Djurić, M.F. Bugallo, Particle filtering, in: T. Adali, S. Haykin (Eds.), *Adaptive Signal Processing*, Wiley, 2010, pp. 271–331.
- [33] A. Marshall, The use of multistage sampling schemes in Monte Carlo computations, in: *Symposium on Monte Carlo*, Wiley, New York, 1956, pp. 123–140.
- [34] T. Kloek, H.K. van Dijk, Bayesian estimates of equation system parameters: an application of integration by Monte Carlo, *Econometrica* 46 (1978) 1–20.
- [35] H.K.V. Dijk, T. Kloek, Further experience in Bayesian analysis using Monte Carlo integration, *Econometrics* 14 (1980) 307–328.
- [36] J. Geweke, Bayesian inference in econometrics models using Monte Carlo integration, *Econometrica* 57 (1989) 1317–1339.
- [37] M.-S. Oh, J. Berger, Adaptive importance sampling in Monte Carlo integration, *J. Stat. Comput. Simul.* 41 (3) (1992) 143–168.
- [38] M.-S. Oh, J. Berger, Integration of multimodal functions by Monte Carlo importance sampling, *J. Am. Stat. Assoc.* 88 (3) (1993) 450–456.
- [39] R. Rubenstein, D. Kroese, *The Cross-Entropy Method. A Unified Approach to Combinatorial Optimization, Monte Carlo Simulation, and Machine Learning*, Springer Verlag, New York, 2004.
- [40] R. Douc, A. Guillin, J.M. Marin, C.P. Robert, Convergence of adaptive mixtures of importance sampling schemes, *Ann. Stat.* 35 (2007) 420–448.
- [41] G. Celeux, J.M. Marin, C.P. Robert, Iterated importance sampling in missing data problems, *Comput. Stat. Data Anal.* 50 (2006) 3386–3404.
- [42] O. Cappé, A. Guillin, J.M. Marin, C.P. Robert, Population Monte Carlo for ion channel restoration, *J. Comput. Graph. Stat.*
- [43] M. Isard, A. Blake, CONDENSATION – conditional density propagation for visual tracking, *Int. J. Comput. Vis.* 29 (1998) 5–28.
- [44] M.C.A.M. Bink, M.P. Boer, C.J.F. Braak, J. Jansen, R.E. Voorrips, W.E. van de Weg, Bayesian analysis of complex traits in pedigreed plant populations, *Euphytica* 161 (2008) 85–96.
- [45] C. Bi, A Monte Carlo EM algorithm for de novo motif discovery in biomolecular sequences, *IEEE/ACM Trans. Comput. Biol. Bioinform.* 6 (2009) 370–386.
- [46] G.E. Barter, L.K. Purvis, N.P. Teclerian, T.H. West, Analysis of detection systems for outdoor chemical or biological attacks, in: *Proceedings of IEEE Conference on Technologies for Homeland Security*, 2009.
- [47] G. Douc, J. Marin, C. Robert, Convergence of adaptive mixtures of importance sampling schemes, *Ann. Stat.* 35 (2007) 420–448.
- [48] G. Douc, J. Marin, C. Robert, Minimum variance importance sampling via population Monte Carlo, *ESAIM Probab. Stat.* 11 (2007) 427–447.
- [49] E. Koblenz, J. Miguez, A population Monte Carlo scheme with transformed weights and its application to stochastic kinetic models, *Stat. Comput.* 25 (2015) 407–425.
- [50] M. West, Approximating posterior distributions by mixtures, *J. R. Stat. Soc. B* 55 (2) (1993) 409–422.
- [51] G.H. Givens, A.E. Raftery, Local adaptive importance sampling for multivariate densities with strong nonlinear relationships, *J. Am. Stat. Assoc.* 91 (443) (1996) 132–141.
- [52] P. Zhang, Nonparametric importance sampling, *J. Am. Stat. Assoc.* 435 (1996) 1245–1253.
- [53] R.J. Steele, A.E. Raftery, M.J. Emond, Computing normalizing constants for finite mixture models via incremental mixture importance sampling (IMIS), *J. Comput. Graph. Stat.* 15 (3) (1996) 712–734.
- [54] M. West, Mixture models, Monte Carlo, Bayesian updating and dynamic models, *Comput. Sci. Stat.* 24 (1993) 325–333.
- [55] L. Pozzi, A. Mira, A R adaptive multiple importance sampling (ARAMIS), document online.
- [56] A. Owen, Y. Zhou, Safe and effective importance sampling, *J. Am. Stat. Assoc.* 95 (2000) 135–143.
- [57] E. Veach, L. Guibas, Optimally combining sampling techniques for Monte Carlo rendering, in: *SIGGRAPH 1995 Proceedings*, 1995, pp. 419–428.
- [58] J. Sirén, P. Marttinen, J. Corander, Reconstructing population histories from single nucleotide polymorphism data, *Mol. Biol. Evol.* 28 (1) (2011) 673–683.
- [59] T. Hesterberg, Weighted average importance sampling and defensive mixture distributions, *Technometrics* 37 (2) (1995) 185–194.
- [60] L. Martino, V. Elvira, D. Luengo, J. Corander, MCMC-driven adaptive multiple importance sampling, in: *Springer Proceedings in Mathematics & Statistics for the 12th Brazilian Meeting on Bayesian Statistics (EBEB)*, Atibaia, Brasil, 2014, pp. 1–13.
- [61] Z.I. Botev, P. L’Ecuyer, B. Tuffin, Markov chain importance sampling with applications to rare event probability estimation, *Stat. Comput.* 23 (2013) 271–285.
- [62] M. Bolić, P.M. Djurić, S. Hong, Resampling algorithms for particle filters: a computational complexity perspective, *EURASIP J. Adv. Signal Process.* 15 (2004) 2267–2277.
- [63] D.B. Rubin, A noniterative sampling/importance resampling alternative to the data augmentation algorithm for creating a few imputations when fractions of missing information are modest: the sir algorithm, *J. Am. Stat. Assoc.* 82 (1987) 543–546.
- [64] E. Eberlein, Application of generalized hyperbolic Lévy motions to finance, in: *Lévy Processes*, Springer, 2001, pp. 319–336.
- [65] E. Eberlein, K. Prause, The generalized hyperbolic model: financial derivatives and risk measures, in: *Mathematical Finance–Bachelier Congress 2000*, Springer, 2002, pp. 245–267.
- [66] M. Abramowitz, I.A. Stegun, Modified Bessel functions I and k, in: *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*, Dover, New York, 1972, pp. 374–377.
- [67] L. Martino, H. Yang, D. Luengo, J. Kanninen, J. Corander, The FUSS algorithm: a fast universal self-tuned sampler within Gibbs, arXiv:1405.0263, 2014, pp. 1–34.

Mónica Bugallo received her B.S., M.S., and Ph.D. degrees in computer science and engineering from University of A Coruña, Spain, in 1996, 1999, and 2001, respectively. In 2002, she joined the Department of Electrical and Computer Engineering at Stony Brook University where she is currently Associate Professor. Her research interests are in the field of statistical signal processing, with emphasis on Bayesian analysis, sequential Monte Carlo methods, adaptive filtering, stochastic optimization, and their applications to different disciplines including communications, biomedicine, target tracking, and vehicle positioning and navigation.

Luca Martino received the Electronic Eng. degree (M.Sc.) in 2006 from Politecnico di Milano (Italy) and the Ph.D. degree in statistical signal processing in 2011 from Universidad Carlos III de Madrid (Spain), where he was assistant professor until July 2013. Since then he has worked as a postdoc researcher at the University of Helsinki (Aug. 2013 – Jan. 2015) and University of Sao Paulo (currently). His research interests include computational statistics and Bayesian inference.

Jukka Corander is a professor of statistics at the University of Helsinki. He received his M.Sc. and Ph.D. degrees in statistics in 1995 and 2000, respectively, from Stockholm University. His main academic interests are graphical models, probabilistic classification theory, stochastic computation theory, statistical models for molecular evolution and population genetics, and data mining with molecular biological data.