Adaptive Importance Sampling: The Past, the Present, and the Future

Mónica F. Bugallo, *Senior Member, IEEE*, Víctor Elvira, *Member, IEEE*, Luca Martino, David Luengo, *Member, IEEE*, Joaquín Míguez, and Petar M. Djurić, *Fello, IEEE*

I. Introduction

A fundamental problem in signal processing is the estimation of unknown parameters or functions from noisy observations. Important examples include localization of objects in wireless sensor networks [1] and the Internet of Things [2]; multiple source reconstruction from electroencephalograms [3]; estimation of power spectral density for speech enhancement [4]; or estimation in genomic signal processing [5]. Within the Bayesian signal processing framework, these problems are addressed by constructing posterior probability distributions of the unknowns. The posteriors combine optimally all the information about the unknowns in the observations with the information that is present in their prior probability distributions. Given the posterior, one often wants to make inference about the unknowns, e.g., if we are estimating parameters, finding the values that maximize their posterior, or the values that minimize some cost function given the uncertainty of the parameters. Unfortunately, obtaining closed-form solutions to these types

M. F. Bugallo and P. M. Djurić are with the Department of Electrical and Computer Engineering, Stony Brook University (USA), e-mail: {monica.bugallo,petar.djuric}@stonybrook.edu. V. Elvira is with IMT Lille Douai, Université de Lille, and CRIStAL laboratory (UMR 9189) (France), e-mail: victor.elvira@telecom-lille-fr. L. Martino is with the Image Processing Laboratory, Universitat de València (Spain), e-mail: luca.martino@uv.es. D. Luengo is with the Department of Signal Theory and Communications, Universidad Politécnica de Madrid (Spain), e-mail: david.luengo@upm.es. J. Míguez is with the Department of Signal Theory and Communications, Universidad Carlos III de Madrid (Spain), e-mail: joaquin.miguez@uc3m.es.

of problems is infeasible in most practical applications, and therefore, developing approximate inference techniques is of utmost interest.

A methodology that comes to the rescue for solving most difficult problems of inference is based on random drawing of samples. It was first applied systematically by the Italian physicist Enrico Fermi when he studied neutron diffusion [6]. However, no publication is available from him on this topic. Later the methodology came to be known as Monte Carlo (MC) sampling.

The MC methods we know today were created by Stanislaw Ulam, John von Neumann and others [7]. Their efforts coincided with the development of the first general computer and resulted in the Metropolis algorithm [8]. The next major advancement of MC methods came with a generalization of the Metropolis algorithm proposed by Hastings in 1970 [9]. All these methods represent a family of simulation-based algorithms that aim at generating samples from a target probability distribution (often a posterior distribution in a Bayesian setting). The algorithms are based on constructing a Markov chain that has the desired distribution as its equilibrium distribution, which is why they are referred to as Markov chain Monte Carlo (MCMC) algorithms [10] (a review of the history of MCMC sampling can be found in [7]). The most prominent MCMC algorithms remain the Metropolis-Hastings and Gibbs sampling algorithms [10]. Since the 1990s, MCMC-based methods have seen tremendous growth and success.

An important alternative to MCMC sampling is the class of Importance Sampling (IS) methods. The IS methods are elegant, theoretically sound, simple-to-understand, and widely applicable [7]. Assume that the aim is to approximate a given *target* probability distribution. The basic IS mechanism consists of (a) drawing samples from simple *proposal* densities, (b) weighting the samples by accounting for the mismatch between the target and the proposal densities, and (c) performing the desired inference using the weighted samples. IS was first used in statistical physics for inference of rare events, and in particular for estimating the probability of nuclear particles that penetrate shields [11]. Later, IS was also used as a variance reduction technique based on

simulating from a proposal density instead of the target density [12]. The interest in IS techniques was running in parallel to the emergence of Bayesian computational methods. The interest was not only driven by their simplicity, but also by their ability to estimate normalizing constants of the target distribution, a feature not shared by MCMC methods that turns out useful in many practical problems (e.g., model selection).

It is well known that the performance of IS methods directly depends on the choice of the proposal densities [7]. When the method is applied naively, only few of the IS weights take relevant values, while the rest are negligible. This phenomenon is widely known in the IS literature as weight degeneracy [7]. If the goal is to estimate the mean of the samples of a target distribution, then the proposals must be adapted to parts of the space where the posterior probability is large, while if the focus is on a problem related to system reliability, then the probability of rare events is better approximated by placing the proposals in the tails of the posterior. Locating the regions from which samples should be drawn may not be easy, which suggests that the main challenge in implementing IS methods lies in finding good proposal densities. However, designing these proposals usually cannot be done a priori, and thus, adaptive procedures must be constructed and applied iteratively. The objective is that with passing iterations the quality of the samples improves and the inference from them becomes more accurate. This leads us to the concept of adaptive importance sampling (AIS). AIS methods are endowed with the nice feature of being able to learn from previously sampled values of the unknowns and consequently, to become more accurate. It is important to note that the AIS algorithms must remain simple, i.e., both the drawing of samples and the computation of their weights should be easily managed.

In this feature article, we first go over the basics of IS and then proceed with explaining the learning process that takes place in AIS and with presenting several state-of-the-art methods. We discuss AIS estimators and their convergence properties, and then show numerical results on signal processing examples. The article also provides an outlook of the research in AIS. For a clearer

presentation, in Table I we display the notation used throughout the paper.

TABLE I: Summary of notation.

Notation	**Description**
d_x	dimension of the unknown parameter vector
$\mathbf{x} \in \mathbb{R}^{d_x}$	unknown realization of a parameter vector
d_y	dimension of the observed data vector
$\mathbf{y} \in \mathbb{R}^{d_y}$	observed data vector
j	iteration variable
J	total number of iterations
N	number of proposal distributions in an iteration
K	number of generated samples per proposal in an iteration
$\tilde{\pi}$	target pdf
$\tilde{\pi}^K$	approximated target pdf with K samples and weights
ℓ	likelihood function
p_0	prior distribution
Z	normalizing constant
$\bar{I}^K$	natural estimator computed from K samples generated from the target
$\widehat{I}^K$	non-normalized estimator computed from K samples
$\widetilde{I}^K$	self-normalized estimator computed from K samples
$\mathbf{x}_{n,j}^{(k)}$	k-th sample of the nth proposal at iteration j
$w_{n,j}^{(k)}$	IS weight associated with $\mathbf{x}_{n,j}^{(k)}$
$\bar{w}_{n,j}^{(k)}$	normalized IS weight associated with $\mathbf{x}_{n,j}^{(k)}$
f	test function/moment of the target
$q_{n,j}$	nth proposal function in the jth iteration
$\boldsymbol{\theta}_{n,j}$	parameters defining the proposal $q_{n,j}$; e.g., $\boldsymbol{\theta}_{n,j} = [\boldsymbol{\mu}_{n,j}\ \mathbf{C}_{n,j}]$ for a Gaussian
$\boldsymbol{\mu}_{n,j}$	location parameter (usually mean) of the proposal $q_{n,j}$
$\mathbf{C}_{n,j}$	scale parameter (usually covariance) of the proposal $q_{n,j}$
$\rho_{n,j}$	weight in the mixture of the nth proposal at iteration j.
∇	gradient
$\mathbf{H}_{\mathbf{x}}$	Hessian evaluated at $\mathbf{x}$
λ_j	gradient step at iteration j
$E_{\tilde{\pi}}[\cdot]$	expected value with respect to the pdf $\tilde{\pi}$

II. Background (with examples)

A. Problem statement

Let us consider a generic inference problem where a d_x-dimensional vector of unknown static real parameters, $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^{d_x}$, has a probability density function (pdf) given by

$$\tilde{\pi}(\mathbf{x}) = \frac{\pi(\mathbf{x})}{Z}, \tag{1}$$

where $\pi(\mathbf{x})$ is a non-normalized non-negative target function, and $Z = \int_{\mathcal{X}} \pi(\mathbf{x}) d\mathbf{x}$ is a finite normalizing constant that may be unknown. The goal is to compute some particular moment of $\mathbf{x}$, which can be defined as

$$I = \int_{\mathcal{X}} f(\mathbf{x}) \tilde{\pi}(\mathbf{x}) d\mathbf{x}, \tag{2}$$

where $f(\cdot)$ can be any function of $\mathbf{x}$ which is integrable with respect to (w.r.t.) $\tilde{\pi}(\mathbf{x})$.

The previous mathematical formulation can be used to represent different problems, including the estimation of rare events [12] or Bayesian inference [7]. For instance, when estimating rare events, Z is perfectly known and the moment of interest can be $f(\mathbf{x}) = \mathbb{I}_{g(\mathbf{x})>0}$, where $g(\mathbf{x})$ is a given function and $\mathbb{I}$ is the indicator function that takes the value 1 if $g(\mathbf{x}) > 0$, and 0 otherwise. In this case, $\tilde{\pi}(\mathbf{x})$ is completely characterized, and the challenge is in computing the integral given by Eq. (2). In Bayesian inference, $\tilde{\pi}(\mathbf{x})$ often represents the posterior distribution that is linked to some observed data, $\mathbf{y} \in \mathbb{R}^{d_y}$, and is expressed as

$$\tilde{\pi}(\mathbf{x}) = p(\mathbf{x}|\mathbf{y}) = \frac{\ell(\mathbf{y}|\mathbf{x}) p_0(\mathbf{x})}{Z(\mathbf{y})} \propto \ell(\mathbf{y}|\mathbf{x}) p_0(\mathbf{x}), \tag{3}$$

where $p(\mathbf{x}|\mathbf{y})$ is the posterior pdf, $\ell(\mathbf{y}|\mathbf{x})$ is the likelihood function, $p_0(\mathbf{x})$ is the prior pdf, and $Z(\mathbf{y})$ is the model evidence or partition function. For some specific statistical models, e.g., when $p_0(\mathbf{x})$ is a conjugate prior of $\ell(\mathbf{y}|\mathbf{x})$ [13], $Z(\mathbf{y}) = \int_{\mathcal{X}} \ell(\mathbf{y}|\mathbf{x}) p_0(\mathbf{x}) d\mathbf{x}$ can readily be obtained. In general, however, computing Z can be a very difficult problem. For this reason, we define the

non-normalized target function

$$\pi(\mathbf{x}) = \ell(\mathbf{y}|\mathbf{x})p_0(\mathbf{x}). \tag{4}$$

From here on and without loss of generality, we focus on the generic case where $Z(\mathbf{y})$ is unknown. To simplify the notation, we drop the dependence of Z on $\mathbf{y}$ and write $Z \equiv Z(\mathbf{y})$. In the rest of the paper, we refer to Z as a normalizing constant. This term is more general than "model evidence" or "marginal likelihood," which are often used in Bayesian theory. Finally, note that we concentrate on real parameters and observations for the sake of clarity in the exposition. However, all of the AIS methods presented and the considerations performed throughout the paper are directly applicable to multidimensional-complex target densities.

B. Monte Carlo methods: motivation and basics

Obtaining closed-form solutions of the described problem is infeasible in most practical applications, and therefore the next best thing is to develop approximate inference techniques with good accuracy. Let us assume that it is possible to draw K independent samples, $\{\mathbf{x}^{(k)}\}_{k=1}^{K}$, from the target distribution $\tilde{\pi}(\mathbf{x})$. The integral I can then be approximated by

$$\bar{I}^K = \frac{1}{K}\sum_{k=1}^{K} f(\mathbf{x}^{(k)}), \qquad \text{where } \mathbf{x}^{(k)} \sim \tilde{\pi}(\mathbf{x}). \tag{5}$$

With the drawn samples, we can approximate the target probability distribution corresponding to the density $\widetilde{\pi}(\mathbf{x})$ as

$$\widetilde{\pi}^K(\mathbf{x}) \;\; = \;\; \frac{1}{K}\sum_{k=1}^{K} \delta(\mathbf{x} - \mathbf{x}^{(k)}), \tag{6}$$

where $\delta(\mathbf{x}-\mathbf{x}^{(k)})$ is the Dirac delta function centered at $\mathbf{x}^{(k)}$. With this approximation, we can estimate I in Eq. (5) by

$$\begin{aligned} I &= \int_{\mathcal{X}} f(\mathbf{x})\tilde{\pi}(\mathbf{x})d\mathbf{x} \\ &\approx \int_{\mathcal{X}} f(\mathbf{x})\tilde{\pi}^K(\mathbf{x})d\mathbf{x} = \frac{1}{K}\sum_{k=1}^{K}\int_{\mathcal{X}} f(\mathbf{x})\delta(\mathbf{x}-\mathbf{x}^{(k)})d\mathbf{x}, \end{aligned} \tag{7}$$

which yields Eq. (5).

The estimator $\bar{I}^K$ is consistent with K, since it converges almost surely to I by the strong law of large numbers [7]. Moreover, it can be easily shown that the estimator is unbiased, i.e., $\mathrm{E}_{\tilde{\pi}}[\bar{I}^K] = I$ and, assuming that $f(\mathbf{x})$ is real and square-integrable, its variance is given by [7]

$$\mathrm{Var}_{\tilde{\pi}}(\bar{I}^K) = \frac{\mathrm{Var}_{\tilde{\pi}}\left(f(\mathbf{X})\right)}{K}. \tag{8}$$

This methodology is known as the *Monte Carlo method* [7], and it was first described in [14].

As already pointed out, very often $\tilde{\pi}(\mathbf{x})$ does not have a known closed form and it is not possible to draw samples from it. Moreover, in some other settings, it might not be convenient to generate samples from the target distribution even if it is possible. This is the case of rare event estimation, where it is not efficient to simulate samples from $\tilde{\pi}(\mathbf{x})$ since the estimation of I would depend on a very low number of *effective* samples [15].

C. Importance sampling: motivation and basics

The IS methodology was first used in statistical physics for rare event inference. More specifically, it was applied to estimate the probability of nuclear particles that penetrate shields [11]. Later, IS was also used as a variance reduction technique based on simulating from a proposal density instead of the target one, reducing the computational effort to compute rare events from the target distribution [12]. The interest in IS techniques has run in parallel to the growth of the theory of Bayesian inference. The reason for this is that often it is not possible to generate samples

from the posterior distribution because it can only be evaluated up to a normalizing constant.

Let us consider K independent samples, $\{\mathbf{x}^{(k)}\}_{k=1}^{K}$, drawn from a single proposal pdf, $q(\mathbf{x})$, with heavier tails than the target, $\pi(\mathbf{x})$. Each sample has an associated importance weight given by

$$w^{(k)} = \frac{\pi(\mathbf{x}^{(k)})}{q(\mathbf{x}^{(k)})}, \quad k = 1, \ldots, K, \tag{9}$$

where the weights represent the significance of the samples in the approximation of the target by the considered proposal. Using the samples and weights, the integral in Eq. (2) can be approximated by a self-normalized estimator as

$$\widetilde{I}^K = \frac{1}{K\widehat{Z}} \sum_{k=1}^{K} w^{(k)} f(\mathbf{x}^{(k)}), \tag{10}$$

where $\widehat{Z} = \frac{1}{K}\sum_{k=1}^{K} w^{(k)}$ is an unbiased estimator of $Z = \int_{\mathcal{X}} \pi(\mathbf{x}) d\mathbf{x}$ [7]. It is not difficult to see that now we approximate the target distribution by

$$\widetilde{\pi}^K(\mathbf{x}) \;\; = \;\; \sum_{k=1}^{K} \bar{w}^{(k)} \delta(\mathbf{x} - \mathbf{x}^{(k)}), \tag{11}$$

where the $\bar{w}^{(k)}$s are normalized weights of the samples obtained by

$$\bar{w}^{(k)} \;\; = \;\; \frac{w^{(k)}}{\sum_{i=1}^{K} w^{(i)}}. \tag{12}$$

If the normalizing constant is known, then it is possible to use the non-normalized estimator

$$\widehat{I}^K = \frac{1}{KZ} \sum_{k=1}^{K} w^{(k)} f(\mathbf{x}^{(k)}). \tag{13}$$

Note that $\widetilde{I}^K$ is only asymptotically unbiased, whereas $\widehat{I}^K$ is unbiased. Both $\widetilde{I}^K$ and $\widehat{I}^K$ are consistent estimators of I and their variance is directly related to the discrepancy between $\tilde{\pi}(\mathbf{x})|f(\mathbf{x})|$ and $q(\mathbf{x})$ [7]. However, when several different moments of the target must be estimated or the function f is unknown *a priori*, a common strategy in IS is to decrease the mismatch between

the proposal $q(\mathbf{x})$ and the target $\tilde{\pi}(\mathbf{x})$ [16]. This is equivalent to minimizing the variance of the weights and consequently the variance of the estimator $\widehat{Z}$.

D. Multiple importance sampling: motivation and basics

The target density can only be evaluated point-wise, and therefore it cannot be easily characterized in many cases. This entails that finding a single good proposal pdf, $q(\mathbf{x})$, is not always possible. A robust alternative consists of using a set of proposal pdfs, $\{q_n(\mathbf{x})\}_{n=1}^N$. The resulting method is referred to as multiple importance sampling (MIS) and it was greatly advanced during the 90s in statistics and computer graphics simulation [12], [17], [18]. MIS constitutes the basis of most of the state-of-the-art AIS algorithms [19], [20], [21], [22], [23], [24].

A general MIS framework has recently been proposed in which different sampling and weighting schemes can be combined [25]. Here, we briefly review the most common sampling and two common weighting schemes. Suppose that we draw one sample from each proposal pdf, i.e.,

$$\mathbf{x}_n \sim q_n(\mathbf{x}), \qquad n = 1, ..., N, \tag{14}$$

where, since $K = 1$, we drop the superscript $^{(k)}$. The most common weighting strategies in the literature are:

1) *Standard MIS* (s-MIS) [19]:

$$w_n = \frac{\pi(\mathbf{x}_n)}{q_n(\mathbf{x}_n)}, \quad n = 1, \ldots, N. \tag{15}$$

2) *Deterministic mixture MIS* (DM-MIS) [18]:

$$w_n = \frac{\pi(\mathbf{x}_n)}{\psi(\mathbf{x}_n)} = \frac{\pi(\mathbf{x}_n)}{\frac{1}{N}\sum_{i=1}^N q_i(\mathbf{x}_n)}, \quad n = 1, \ldots, N, \tag{16}$$

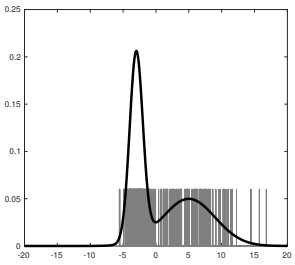
where $\psi(\mathbf{x})$ represents the mixture pdf composed of all the proposal pdfs evaluated at $\mathbf{x}$.

From the weighted set $\{\mathbf{x}_n, w_n\}_{n=1}^N$, generated by either the s-MIS or the DM-MIS methods

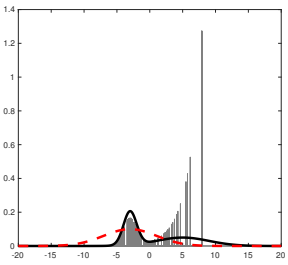
described above, we can compute a self-normalized estimator $\widetilde{I}^N$ and a non-normalized estimator $\widehat{I}^N$ in the same way as in Eqs. (10) and (13), respectively. The self-normalized $\widetilde{I}^N$ is consistent and asymptotically unbiased, whereas the non-normalized $\widehat{I}^N$ is both consistent and unbiased. The DM approach is superior w.r.t. that of s-MIS in terms of variance of the estimator $\widehat{I}^N$, as proved in [25]. Although both alternatives perform the same number of target evaluations, the DM estimator is computationally more expensive w.r.t. the number of proposal evaluations. In particular, s-MIS and DM require N and N^2 evaluations, respectively. Therefore, in scenarios where the number of proposals N is large, the $\mathcal{O}(N^2)$ in the number of proposal evaluations can be prohibitive. Alternative efficient solutions have recently been devised to mitigate this excess of computational load [26], [27].

Figure 1 illustrates the processes of sampling and weighting based on the different methods explained in this section. More specifically, Fig. 1(a) displays the generated samples and associated weights when sampling from the target distribution *is possible*. We observe that all the weights are equal in this case. For both Figs. 1(b) and 1(c), the generation of samples is performed using a single proposal pdf. However, the proposal pdfs, plotted with dashed lines, are differently located, and therefore one can appreciate how the second choice is more appropriate by observing the variability of the weight values. Note that the scale of the vertical axes is different in order to show the large weights in Fig. 1(b). Figures 1(d) and 1(e) use the concept of MIS, i.e., there we use two proposal pdfs. The weights in Fig. 1(d) are calculated using the standard formulation of weight update from Eq. (15), while in Fig. 1(e), they are computed according to Eq. (16). It is clear that a smaller variance of the weights is achieved with the DM approach.

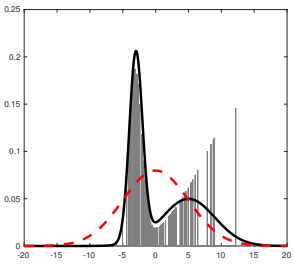
Finally, the validity of the possible different weighting schemes for MIS is justified in [25] by using the concept of a proper set of weighted samples. More precisely, the suitability of a particular MIS scheme is guaranteed if the non-normalized estimator $\widehat{I}^N$ and the normalizing constant estimator $\widehat{Z}$ are unbiased and consistent, which also implies that the self-normalized



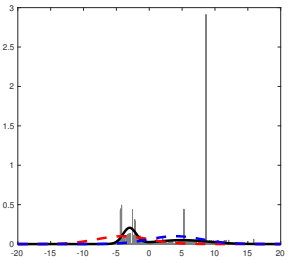
(a) MC sampling directly from the target.



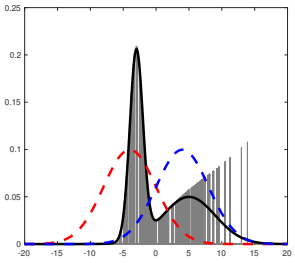
(b) IS, single proposal pdf.



(c) IS, single proposal pdf (with a better location than that in (b)).



(d) MIS with standard weights.



(e) MIS with DM weights.

Fig. 1: Approximations of the target pdf, $\pi(\mathbf{x})$, by different discrete probability distributions (displayed by thin bars with weights corresponding to heights of the bars). The target pdfs are shown by solid lines, while the proposal pdfs are plotted with dashed lines. (a) Ideal situation: an approximation with equally weighted samples, as they are drawn directly from the target. (b)–(c) Approximations with IS and a single proposal to show the effect of the location: a better proposal placement leads to more uniform weights. (d)–(e) Approximations with MIS and two proposals to show the effect of the choice of the weighting scheme: the deterministic mixture (DM) approach leads to more uniform weights than the standard approach.

estimator $\widetilde{I}^N$ is consistent.

III. Adaptive Importance Sampling

A. The basics of AIS

The AIS methodology is based on an iterative process for gradual evolution of the single or multiple proposal densities to accurately approximate the target pdf. The procedure consists of three basic steps: generation of samples from a proposal or set of proposals (sampling), calculation of the importance of each of the samples (weighting), and updating (adapting) the parameters that define the proposal(s) to obtain the new proposal(s) for the next iteration. Figure 2 shows a simple

flow diagram of the steps of AIS with only one proposal pdf. The diagram also shows the possible data dependencies among the basic steps.

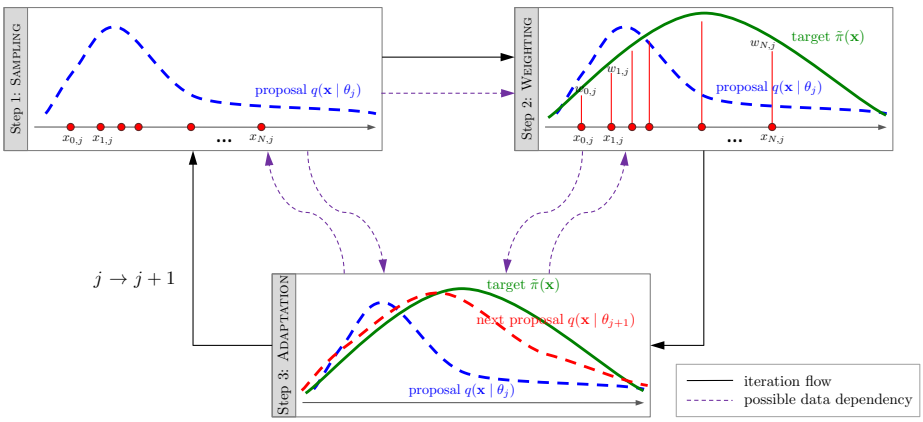


Fig. 2: A generic flow diagram of the AIS methodology, showing the three steps that must be performed iteratively by any AIS algorithm (sampling, weighting and adaptation), and the data flow among these steps.

In the general case, the algorithm is initialized with a set of N proposals $\{q_n(\mathbf{x}|\boldsymbol{\theta}_{n,1})\}_{n=1}^{N}$, each one parametrized by a vector $\boldsymbol{\theta}_{n,1}$. After drawing a set of samples, $\mathbf{x}_{n,1}^{(k)}$, $n = 1, \ldots, N, k = 1, \ldots, K$ (recall that K is the number of samples generated by a proposal), and weighting them, one obtains a discrete probability distribution that approximates the target distribution, $\{x_{n,1}^{(k)}, w_{n,1}^{(k)}\}$, $n = 1, \ldots, N, k = 1, \ldots, K$. Then, the parameters of the n-th proposal are updated from $\boldsymbol{\theta}_{n,1}$ to $\boldsymbol{\theta}_{n,2}$. This process is repeated, i.e., sampling, weighting and moving from $\boldsymbol{\theta}_{n,j}$ to $\boldsymbol{\theta}_{n,j+1}$, until an iteration stoppage criterion is met (e.g., a maximum number of iterations, J, is reached). Table II outlines the main steps of the general algorithm.

Figure 3 shows the evolution in the approximation of a target pdf, $\tilde{\pi}(\mathbf{x})$, which in this case is a mixture of two Gaussian pdfs. In this example just one Gaussian proposal ($N = 1$) is used, $q_1(\mathbf{x})$, with initial vector parameter $\boldsymbol{\theta}_{1,1} = [\mu_1 \ \sigma_1^2] = [-4 \ 3]$, where μ_1 and σ_1^2 denote the mean and the variance, respectively. Figure 3 displays three iterations of the AIS algorithm, where the initial parameter vector $\boldsymbol{\theta}_{1,1}$ is updated in the next proposal so that it can produce samples and weights that yield a better approximation of the target distribution. Note that the final scale and location

TABLE II: Generic AIS algorithm

Initialization
Choose K, N, J, $\{\boldsymbol{\theta}_{n,1}\}_{n=1}^{N}$
For $j = 1, \ldots, J$:
1. Sampling
Draw K samples from each of the N proposal pdfs, $\{q_{n,j}(\boldsymbol{\theta}_{n,j})\}_{n=1}^{N}$, $\mathbf{x}_{n,j}^{(k)}, k = 1, \ldots, K,\ n = 1, \ldots, N$
2. Weighting
Calculate the weights, $w_{n,j}^{(k)}$, for each of the generated KN samples.
3. Adaptation
Update the proposal parameters $\{\boldsymbol{\theta}_{n,j}\}_{n=1}^{N} \longrightarrow \{\boldsymbol{\theta}_{n,j+1}\}_{n=1}^{N}$.
Outputs
Return the KNJ pairs $\{\mathbf{x}_{n,j}^{(k)}, w_{n,j}^{(k)}\}$ for all $k = 1, \ldots, K$, $n = 1, \ldots, N$, $j = 1, \ldots, J$.

of the proposal is much more adequate than the starting proposal in that it effectively covers both modes of the target.

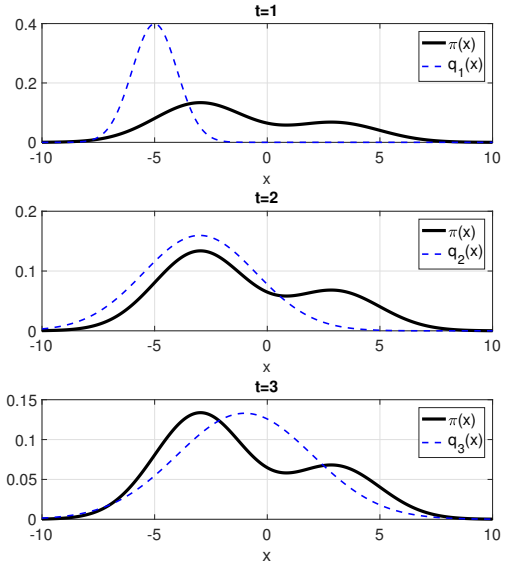


Fig. 3: Proposal adaptation through AIS. The initial proposal $q_1(x)$ (too narrow and poorly placed) is iteratively moved towards a better location at some intermediate location between the two modes of the target pdf and widened in order to properly cover the effective support of the target.

In order to approximate the integral I in Eq. (2), there exist different possibilities for combining all the KNJ weighted samples, $\{\mathbf{x}_{n,j}^{(k)}, w_{n,j}^{(k)}\}$, generated by the AIS method [28]. A common (and straightforward) choice is to assign to each sample a normalized weight $\bar{w}_{n,j}^{(k)}$, which considers all

the weights, i.e.,

$$\bar{w}_{n,j}^{(k)} = \frac{w_{n,j}^{(k)}}{\sum_{l=1}^{J}\sum_{i=1}^{N}\sum_{r=1}^{K} w_{i,l}^{(r)}}. \tag{17}$$

Hence, the self-normalized AIS estimator is $\widetilde{I}^{KNJ} = \sum_{j=1}^{J}\sum_{n=1}^{N}\sum_{k=1}^{K} \bar{w}_{n,j}^{(k)} f(\mathbf{x}_{n,j}^{(k)})$.

B. Modern AIS methods

AIS methods got their turn in the spotlight of MC computations after the publication of the population Monte Carlo (PMC) sampling method by Cappé et al. in 2004 [19], notwithstanding the existence of several AIS schemes at that time (see [28] for a review). The PMC methodology offered a framework to adapt a population of proposals which was simple, flexible and free from the convergence and ergodicity issues of adaptive MCMC techniques. The original PMC algorithm used a multinomial resampling stage (note that any of the better alternative resampling strategies developed for particle filters can also be used [29]) and was unstable due to the use of the s-MIS weighting strategy of Eq. (15). However, the proposed approach raised a considerable interest within the computational statistics community, and improved PMC algorithms shortly followed, like the D-kernel PMC [30], [31] or the mixture PMC (M-PMC) [20]. Furthermore, several authors have recently shown that the performance of PMC can be improved even more through the use of a nonlinear transformation of the weights [32] or the combination of the DM weighting scheme of Eq. (16) and sophisticated resampling schemes [24].

On the other hand, encouraged by the renewed interest in AIS methods spurred by the PMC approach, several authors have proposed AIS algorithms that do not fall within the PMC framework. For instance, the idea of incremental IS mixtures (originally proposed in [33]) was taken up again by Cornuet et al. in the adaptive multiple importance sampling (AMIS) method [21]. AMIS uses a single proposal per iteration, but applies the DM weighting scheme of Eq. (16) using a mixture composed of the present and all past proposal pdfs. Much more robust and stable estimators are thus obtained, but at the expense of a substantial increase in the computational

cost. An alternative to AMIS is the recently proposed adaptive population importance sampling (APIS) algorithm [22]. APIS is also based on the DM weighting scheme of Eq. (16), but it uses a mixture with a fixed number of proposals per iteration. In this way, APIS inherits the robustness and stability of AMIS, but with the benefit of allowing a user controllable computational cost that does not increase as the algorithm is iterated. Moreover, gradient information can be incorporated to the APIS algorithm in order to improve the performance in high-dimensional state spaces [34].

Finally, note that the combination of MCMC and AIS techniques has also been considered in several works. For instance, MCMC steps can be used to accelerate the adaptation of the AIS technique [22], or the MCMC outputs can be used to build a proposal distribution for AIS estimation [35]. Sequential MC samplers have also been suggested as AIS schemes in static scenarios [36].

IV. Implementation and classification of AIS algorithms

A. Implementation of AIS algorithms

Many important AIS algorithms have been proposed in the literature in the last two decades. In this section we describe in detail some of the most popular AIS algorithms:

- Standard population Monte Carlo (PMC) [19]: In this algorithm, N proposals are adapted via resampling, which is a well-known mechanism in MC methodologies that allows us to select the most promising samples and to eliminate those with low weights in order to avoid particle degeneracy [29]. At each iteration, exactly one sample is drawn from each proposal and weighted with the standard IS weights calculated by Eq. (15). Then, N multinomial resampling steps (with replacement) are performed within the population of the N drawn samples (one sample is generated per proposal, i.e., $K = 1$). The surviving set of particles constitutes the set of location parameters for the next population of proposals.

- Mixture population Monte Carlo (M-PMC) [20]: For this method, the proposal used to generate K samples at each iteration is a mixture of N kernels, where the mixture is adapted to decrease the Kullback-Leibler divergence between the mixture and the target. In its simplest version, the algorithm adapts the location, scale and weight of each kernel in the mixture.
- Nonlinear population Monte Carlo (N-PMC) [32]: In this algorithm, the weights are computed in two steps. First, standard importance weights $w_j^{(k)}$ are obtained. Then, a nonlinear function is applied to calculate a set of transformed weights $\breve{w}_j^{(k)}$. The goal of this transformation is to reduce the variance of the weights and hence avoid, or at least mitigate, the weight degeneracy problem. While the standard weights can be used for estimation, the nonlinearly-transformed weights are crucially used for the adaptation step. The latter can be carried out in different ways, with [32] advocating for a simple Gaussian proposal where both the mean vector and the covariance matrix are adapted through the iterations.
- Layered adaptive importance sampling (LAIS) [23]: The adaptive process of the LAIS algorithm is independent of the samples drawn at each iteration. In particular, the algorithm can be seen as a two-layer procedure where the location parameters of the proposals are adapted through one or several MCMC steps with the target as the stationary distribution. In its basic version, a single MCMC step is independently performed at each location parameter.
- Deterministic mixture population Monte Carlo (DM-PMC) [24]: This algorithm meets the simplicity of the standard PMC of [19] with a very high performance. DM-PMC calculates the weights using Eq. (16) instead of Eq. (15), which provides two important advantages, namely the variance of the estimators is decreased (see [25]) and the resampling step with the DM weights promotes the replication of proposals in relevant parts of the target that are underrepresented by the set of proposals (i.e., the exploration is coordinated). DM-PMC generates K samples per each of the N proposals (instead of one, as in [19]). At each iteration, the population of KN samples must be reduced to N via either global or local

resampling.

- Adaptive multiple importance sampling (AMIS) [21]: In this algorithm, just one proposal is used and adapted over the iterations. The adaptive procedure consists of estimating the moments of the target with the available set of K weighted samples, and fitting the moments of the proposal. Its key feature is the re-weighting of all the past samples with a *temporal* mixture weight where the whole sequence of proposals is used in the denominator.
- Gradient adaptive population importance sampling (GAPIS) [34]: Similarly to the LAIS algorithm, GAPIS adapts N proposals by a process that is independent of the samples. In its basic version, the location parameters of the proposals are adapted via a gradient ascent of the target, and the scale parameter by using the Hessian of the target. An advanced implementation is proposed which adds a repulsive interaction among proposals to promote a cooperative exploration of the target.

In Tables III and IV, six out of the seven previous algorithms are outlined by means of pseudocodes. Note that we follow the structure sampling-weighting-adaptation described in Fig. 2 and Table II. We have skipped the N-PMC scheme in these tables for the sake of clarity. We simply point out that, in this algorithm, the standard weights $w_{n,j}^{(k)}$ are transformed using a nonlinearity Φ, namely $\breve{w}_{n,j}^{(k)} = \Phi\left(k, \{w_{n,j}^{(l)}\}_{l=1}^{K}\right)$. These transformed weights are then fed to the adaptation stage. In [32], the nonlinearity $\Phi(\cdot,\cdot)$ is either a tempering or a simple truncation of the largest weights, while the adaptation is carried out as in the AMIS method of Table IV.

B. Classification of relevant AIS algorithms

Table V serves as a summary and compares the main features of different AIS implementations. The features include the number of proposals, the weighting procedure, the updating strategy of the parameters, and the updated parameters. Note that most of the algorithms use more than one proposal. However, due to the adaptive procedure, even with $N = 1$, more than one proposal

TABLE III: Pseudocodes of PMC, DM-PMC, and LAIS.

PMC	DM-PMC	LAIS
Initialization		
J, N, $K = 1$ $\{\boldsymbol{\theta}_{n,1}\}_{n=1}^{N} \equiv \{\boldsymbol{\mu}_{n,1}, \mathbf{C}_n\}_{n=1}^{N}$	J, N, K, $\{\boldsymbol{\theta}_{n,1}\}_{n=1}^{N} \equiv \{\boldsymbol{\mu}_{n,1}, \mathbf{C}_n\}_{n=1}^{N}$	J, N, K, $\{\boldsymbol{\theta}_{n,1}\}_{n=1}^{N} \equiv \{\boldsymbol{\mu}_{n,1}, \mathbf{C}_n\}_{n=1}^{N}$
For $j = 1, \ldots, J$:		
1. Sampling		
$\mathbf{x}_{n,j} \sim q_{n,j}(\mathbf{x}\|\boldsymbol{\mu}_{n,j}, \mathbf{C}_n)$ $n = 1, \ldots, N$	$\mathbf{x}_{n,j}^{(k)} \sim q_{n,j}(\mathbf{x}\|\boldsymbol{\mu}_{n,j}, \mathbf{C}_n)$ $n = 1, \ldots, N$ $k = 1, \ldots, K$	$\mathbf{x}_{n,j}^{(k)} \sim q_{n,j}(\mathbf{x}\|\boldsymbol{\mu}_{n,j}, \mathbf{C}_n)$ $n = 1, \ldots, N$ $k = 1, \ldots, K$
2. Weighting		
$w_{n,j} = \frac{\pi(\mathbf{x}_{n,j})}{q_{n,j}(\mathbf{x}_{n,j})}$ $n = 1, \ldots, N$	$w_{n,j}^{(k)} = \frac{\pi(\mathbf{x}_{n,j}^{(k)})}{\frac{1}{N}\sum_{i=1}^{N} q_{i,j}(\mathbf{x}_{n,j}^{(k)})}$ $n = 1, \ldots, N$ $k = 1, \ldots, K$	$w_{n,j}^{(k)} = \frac{\pi(\mathbf{x}_{n,j}^{(k)})}{\frac{1}{N}\sum_{i=1}^{N} q_{i,j}(\mathbf{x}_{n,j}^{(k)})}$ $n = 1, \ldots, N$ $k = 1, \ldots, K$
3. Adaptation		
Multinomial resampling with replacement over $\{\mathbf{x}_{n,j}, \bar{w}_{n,j} = \frac{w_{n,j}}{\sum_{i=1}^{N} w_{i,j}}\}_{n=1}^{N}$ to update $\{\boldsymbol{\mu}_{n,j+1}\}_{n=1}^{N}$	Multinomial resampling with replacement over $\{\mathbf{x}_{n,j}^{(k)}, \bar{w}_{n,j}^{(k)} = \frac{w_{n,j}^{(k)}}{\sum_{j=1}^{N}\sum_{m=1}^{K} w_{j,j}^{(m)}}\}_{n=1,k=1}^{N,K}$ to update $\{\boldsymbol{\mu}_{n,j+1}\}_{n=1}^{N}$	One (or more) MCMC steps from $\boldsymbol{\mu}_{n,j}$ to $\boldsymbol{\mu}_{n,j+1}$, with $\tilde{\pi}$ as a stationary distribution, for $n = 1, ..., N$
Outputs		
$\{\mathbf{x}_{n,j}, w_{n,j}\}$ $n = 1, \ldots, N$ $j = 1, \ldots, J$	$\{\mathbf{x}_{n,j}^{(k)}, w_{n,j}^{(k)}\}$ $n = 1, \ldots, N$ $k = 1, \ldots, K$ $j = 1, \ldots, J$	$\{\mathbf{x}_{n,j}^{(k)}, w_{n,j}^{(k)}\}$ $n = 1, \ldots, N$ $k = 1, \ldots, K$ $j = 1, \ldots, J$

is used. This is exploited in AMIS and in some implementations of LAIS, where the *temporal* mixture of proposals is used to re-weight the samples via deterministic mixture IS weights. Note that the different adaptive mechanisms can be classified into mechanism based on (a) resampling, (b) moment matching, and (c) independent adaptive processes. Moreover, the moment matching can include all the past weighted samples (AMIS) or just those of the current iteration (APIS). Figure 4 shows three possible dependence charts related to generated samples and the adaptation of the proposal parameters. Note also that, although all the proposal parameters can be adapted,

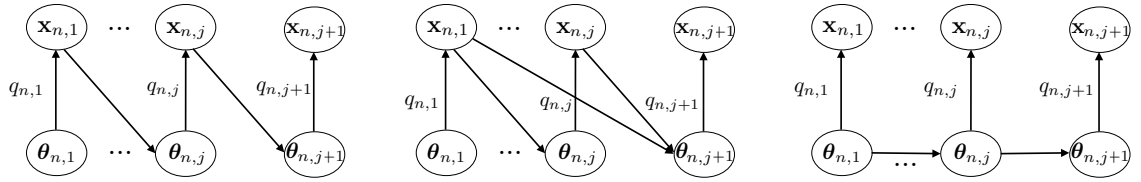
TABLE IV: Pseudocodes of AMIS, GAPIS, and M-PMC.

AMIS	GAPIS	M-PMC
Initialization		
$J,\ K,\ N=1,\ \boldsymbol{\theta}_1 \equiv \{\boldsymbol{\mu}_1, \mathbf{C}_1\}$	$J,\ N,\ K,$ $\{\boldsymbol{\theta}_{n,1}\}_{n=1}^{N} \equiv \{\boldsymbol{\mu}_{n,1}, \mathbf{C}_n\}_{n=1}^{N}$	$J,\ N,\ K,\ \{\boldsymbol{\theta}_{n,1}\}_{n=1}^{N} \equiv$ $\{\rho_{n,1}, \boldsymbol{\mu}_{n,1}, \mathbf{C}_{1,n}\}_{n=1}^{N}$
For $j = 1, \ldots, J$**:**		
1. Sampling		
$\mathbf{x}_j^{(k)} \sim q_j(\mathbf{x}\|\boldsymbol{\mu}_j, \mathbf{C}_j)$ $k = 1, \ldots, K$	$\mathbf{x}_{n,j}^{(k)} \sim q_{n,j}(\mathbf{x}\|\boldsymbol{\mu}_{n,j}, \mathbf{C}_n)$ $n = 1, \ldots, N$ $k = 1, \ldots, K.$	$\mathbf{x}_j^{(k)} \sim \sum_{i=1}^{N} \rho_{i,j} q_{i,j}(\mathbf{x}\|\boldsymbol{\mu}_{i,j}, \mathbf{C}_{i,j}),$ $k = 1, \ldots, K.$
2. Weighting		
$w_j^{(k)} = \frac{\pi(\mathbf{x}_j^{(k)})}{\frac{1}{j}\sum_{i=1}^{j} q_i(\mathbf{x}_j^{(k)})}$ $k = 1, \ldots, K$	$w_{n,j}^{(k)} = \frac{\pi(\mathbf{x}_{n,j}^{(k)})}{\frac{1}{N}\sum_{i=1}^{N} q_{i,j}(\mathbf{x}_{n,j}^{(k)})}$ $n = 1, \ldots, N$ $k = 1, \ldots, K$	$w_j^{(k)} = \frac{\pi(\mathbf{x}_j^{(k)})}{\sum_{i=1}^{N} \rho_{i,j} q_{i,j}(\mathbf{x}_j^{(k)})}$ $k = 1, ..., K$
3. Adaptation		
Update $\boldsymbol{\mu}_{j+1}$ and $\mathbf{C}_{j+1}$ with the empirical mean and covariance using all the weighted samples	Use a suitable λ_j to update $\boldsymbol{\mu}_{n,j+1} =$ $\boldsymbol{\mu}_{n,j} + \lambda_j \nabla \log\left(\pi(\boldsymbol{\mu}_{n,j})\right)$ and the Hessian matrix of $-\log(\pi(\mathbf{x}))$ to update $\mathbf{C}_{n,j+1} = \left(\mathbf{H}_{\boldsymbol{\mu}_{n,j}}\right)^{-1}$	Update $\{\rho_{n,j+1}, \boldsymbol{\mu}_{n,j+1}, \mathbf{C}_{n,j+1}\}_{n=1}^{N}$ by minimizing the KL distance between the proposal and the target approximation
Outputs		
$\{\mathbf{x}_j^{(k)}, w_j^{(k)}\}$ $k = 1, \ldots, K$ $j = 1, \ldots, J$	$\{\mathbf{x}_{n,j}^{(k)}, w_{n,j}^{(k)}\}$ $n = 1, \ldots, N$ $k = 1, \ldots, K$ $j = 1, \ldots, J$	$\{\mathbf{x}_j^{(k)}, w_j^{(k)}\}$ $k = 1, \ldots, K$ $j = 1, \ldots, J$

in the basic implementation of most algorithms, just the location parameters are adapted.

TABLE V: Comparison of various AIS algorithms according to different features.

Algorithm	**# proposals**	**Weighting**	**Adaptation strategy**	**Parameters adapted**
Standard PMC	$N > 1$	standard	resampling	location
M-PMC	$N > 1$	spatial mixture	resampling	location
N-PMC	either	nonlinear	moment estimation	location/scale
LAIS	$N > 1$	generic mixture	MCMC	location
DM-PMC	$N > 1$	spatial mixture	resampling	location
AMIS	$N = 1$	temporal mixture	moment estimation	location/scale
GAPIS	$N > 1$	spatial mixture	gradient process	location/scale
APIS	$N > 1$	spatial mixture	moment estimation	location



(a) The proposal parameters are adapted using the last set of drawn samples (Standard PMC, DM-PMC, N-PMC, M-PMC, APIS).

(b) The proposal parameters are adapted using all drawn samples up to the latest iteration (AMIS).

(c) The proposal parameters are adapted using an independent process from the samples (LAIS, GAPIS).

Fig. 4: Graphical description of three possible dependencies between the adaptation of the proposal parameters $\boldsymbol{\theta}_{n,t}$ and the samples. Note that $q_{n,t} \equiv q_{n,t}(\mathbf{x}|\boldsymbol{\theta}_{n,t})$.

Table VI provides a comparison of the computational complexity of the different algorithms. We display the number of target and proposal evaluations, and also the same quantities per drawn sample. We observe that in AMIS the number of proposal evaluations is increased with the number of iterations, while in the algorithms with deterministic mixture weights this problem appears when we increase the number of proposals. In the latter case, the strategies proposed in [26], [27] can be employed to reduce the number of proposal evaluations. Although this is not displayed in Table VI, the GAPIS algorithm also requires NJ gradient and Hessian evaluations in total, i.e., one per proposal at each iteration.

TABLE VI: Comparison of various AIS algorithms according to the computational complexity.

Algorithm	**# target eval**	**# proposal eval**	**# target eval/sample**	**# proposal eval/sample**
Standard PMC	NJ	NJ	1	1
N-PMC	NJ	NJ	1	1
M-PMC	KJ	KNJ	1	N
LAIS	$K(N+1)J$	KN^2J	$1+1/N$	N
DM-PMC	KNJ	KN^2J	1	N
AMIS	KJ	KJ^2	1	J
GAPIS	KNJ	KN^2J	1	N
APIS	KNJ	KN^2J	1	N

C. A brief summary and comparison of AIS algorithms

In this section, we provide intuition behind the relevant AIS algorithms presented above. The standard PMC [19] opened the door for the fast growth of the AIS methodology. While the

simplicity is its main advantage, the use of the standard IS weights of Eq. (15) has two adverse effects: (a) the variance of the estimators is increased, and (b) each importance weight measures the difference between the target and a specific proposal (regardless of where the other $N-1$ proposals are placed). The latter effect precludes a stable and coordinated adaptation of the whole mixture of proposals, and provokes a path degeneracy due to the resampling step.

The M-PMC [20] addresses the weak points of the standard PMC by applying a robust Rao-Blackwellization step in the adaptation of the proposals. The goal in M-PMC is to iteratively decrease the KL divergence between the target and the mixture of proposals (for the first time, they are seen as a mixture instead of a collection of proposals). M-PMC is more robust and allows for the adaptation of the covariance of each proposal and its weight in the mixture. The disadvantage is the extra computational cost and the potential instability in the adaptation of the covariance (it can tend to a delta) and in the mixture weights (the mixture can end up being formed by just one proposal).

The DM-PMC addresses the open challenges of the standard PMC in a different way. The use of deterministic mixture IS weights, followed by the resampling step, implicitly aims at iteratively reducing the mismatch between the target and the mixture of proposals (see Eq. (16)). In addition, DM-PMC allows to draw $K > 1$ samples per proposal per iteration, which improves the local exploration in the region of each proposal and then increases the stability of the algorithm. Two variants of the algorithm, GR-PMC and LR-PMC, allow for different resampling steps to transition from NK samples in iteration j to N proposals in iteration $j+1$. The advantage of DM-PMC and its variants is the simplicity in the implementation and the high performance. The disadvantage is that only the location parameter of the proposals is adapted.

In general, all the PMC-based algorithms use the set of weighted samples to adapt the proposals. While this recycling is efficient, the dependence between the samples and the next generation of proposals hinders the theoretical analysis of the algorithms.

The LAIS algorithm disconnects the sampling and the adaptive procedures by establishing a two-layer scheme (see Fig. 4(c)). In its simplest version, the adaptive layer of LAIS is driven by Metropolis-Hastings chains, enjoying some of the advantages of the MCMC methods, e.g., their good behavior in high dimension. The LAIS scheme is simple and shows good performance, but again it does not adapt the covariance of the proposals.

The GAPIS algorithm also decouples the adaptation and sampling procedures, adding the information of the gradient and Hessian of the target in the adaptation of the proposals. This scheme performs well in challenging problems, even in high dimensions, and is able to adapt the location and scale parameters of the proposals. Its main disadvantage is the complexity associated to the computation of the gradient and the Hessian.

The AMIS algorithm is also simple because the proposal adaptation is carried out via moment matching. The algorithm has shown good performance in a variety of problems. Furthermore, it is robust because the IS weights are permanently recomputed via Rao-Blackwellization by using the deterministic mixture idea with the mixture of temporal proposals. The main disadvantage is precisely this recomputation of all the weights at every iteration, which precludes its use when the needed number of iterations J is high. The DM-PMC, LAIS and GAPIS methods are particularly well-suited to multimodal target distributions, which are often hard for conventional algorithms (e.g., non-adaptive importance samplers or classical MCMC schemes).

Finally, note that the nonlinear transformation of the importance weights featured by the NPMC method (in order to reduce the weight variance) can readily be applied to other schemes (DM-PMC, AMIS, etc.). This is especially useful at the first stages of the adaptation, when the proposal(s) can still be poorly aligned with the target density and the use of transformed weights can often prevent severe sample impoverishment. Once the the proposal is roughly adapted, the nonlinear transformation can be dropped and conventional weights can be used to reduce the computational cost.

V. Discussion of AIS methods

A. Convergence of IS estimators

The convergence of IS schemes is often assessed in terms of the approximation of integrals of test functions. Specifically, if $\mathbf{X}$ is a random vector of interest, taking values on $\mathbb{R}^{d_x}$ and with pdf $\widetilde{\pi}(\mathbf{x})$, then we study the approximation of the integral

$$I(f) \quad = \quad \int_{\mathcal{X}} f(\mathbf{x})\widetilde{\pi}(\mathbf{x})d\mathbf{x}, \tag{18}$$

where $f : \mathbb{R}^{d_x} \to \mathbb{R}$ is a real test function, assumed integrable w.r.t. the density $\widetilde{\pi}(\mathbf{x})$ (now we make the test function f explicit in the notation). Note that $I(f)$ is the expected value of the real random variable $f(\mathbf{X})$, which can be alternatively denoted by $E_{\widetilde{\pi}}[f(\mathbf{X})]$, and the integrability assumption simply states that this expectation exists, i.e., $E_{\widetilde{\pi}}[f(\mathbf{X})] < \infty$.

We recall that a standard IS scheme with a proposal function $q(\mathbf{x})$ produces a set of random weighted samples $\{\mathbf{x}^{(k)}, w^{(k)}\}_{k=1}^{N}$, where $\mathbf{x}^{(k)} \sim q(\mathbf{x})$ and $w^{(k)} = \frac{\pi(\mathbf{x}^{(k)})}{q(\mathbf{x}^{(k)})}$, that we use to approximate the integral $I(f)$ as

$$\widetilde{I}^K(f) \quad = \quad \frac{1}{\sum_{i=1}^{K} w^{(i)}} \sum_{k=1}^{K} w^{(k)} f(\mathbf{x}^{(k)}). \tag{19}$$

Note that $\widetilde{I}^K(f)$ is a random variable itself. Intuitively, we expect that the error $I(f) - \widetilde{I}(f)$ should vanish, in some proper probabilistic sense, when $K \to \infty$. This is, indeed, a consequence of the strong law of large numbers [7]. Assuming that $q(\mathbf{x}) > 0$ whenever $\pi(\mathbf{x}) > 0$, it can be proved that [37]

$$\lim_{K\to\infty} \widetilde{I}^K(f) = I(f) \quad \text{almost surely (a.s.)}, \tag{20}$$

and it is said that $\widetilde{I}^K(f)$ is a consistent estimator of $I(f)$. Under additional, yet mild, assumptions

on the weight and test functions, namely,

$$E_{\widetilde{\pi}}[w(\mathbf{X})] < \infty \quad \text{and} \quad E_{\widetilde{\pi}}[f^2(\mathbf{X})w(\mathbf{X})] < \infty, \tag{21}$$

a central limit theorem (CLT) also holds for the IS estimator [37]. (Note that here we use the notation $w(\mathbf{X})$ to remind the reader that the weights are functions of the random vector $\mathbf{X}$ and therefore are random variables themselves.) In particular,

$$\sqrt{K}\left(\widetilde{I}^K(f) - I(f)\right) \stackrel{d}{=} \mathcal{N}(0, \sigma^2(f)), \tag{22}$$

where $\stackrel{d}{=}$ denotes convergence of the limit in distribution and the limit variance depends on the test function, namely $\sigma^2(f) \propto E_{\widetilde{\pi}}\left[\left(f(\mathbf{X}) - E_{\widetilde{\pi}}[f(\mathbf{X})]\right)^2 w(\mathbf{X})\right]$.

Equation (22) is one of various results that show how IS estimators converge with the *optimal* Monte Carlo rate $\mathcal{O}\left(\frac{1}{\sqrt{K}}\right)$, i.e., the errors are asymptotically of the same order as with the standard Monte Carlo estimator constructed with K i.i.d. samples from the target pdf $\widetilde{\pi}(\mathbf{x})$. The same optimal rate is obtained for the convergence of the L_p norms of the errors $\widetilde{I}^K(f) - I(f)$ if we assume that both the test function f and the weight function w are bounded, namely

$$\|f\|_\infty = \sup_{\mathbf{x}\in\mathbb{R}^{d_x}} |f(\mathbf{x})| < \infty \quad \text{and} \quad \|w\|_\infty = \sup_{\mathbf{x}\in\mathbb{R}^{d_x}} |w(\mathbf{x})| = \sup_{\mathbf{x}\in\mathbb{R}^{d_x}} \left|\frac{\pi(\mathbf{x})}{q(\mathbf{x})}\right| < \infty, \tag{23}$$

where $\|Z\|_p$ indicates the L_p norm of the random variable Z with a pdf $g(z)$, i.e., $\|Z\|_p = \left(\int Z^p g(z)dz\right)^{\frac{1}{p}}$. Whenever Eq. (23) holds, it can be proved that [38]

$$\|I(f) - \widetilde{I}^K(f)\|_p \leq \frac{c\|f\|_\infty}{\sqrt{K}}, \tag{24}$$

for any $p \geq 1$ and some constant $c < \infty$ independent of K. The inequality in Eq. (24) is easily extended, using a standard argument based on the Markov inequality and the Borel-Cantelli lemma [39], to yield $\lim_{K\to\infty} \widetilde{I}^K(f) = I(f)$ a.s.

A more sophisticated analysis allows us to obtain an upper bound for the random error (not

just for its L_p norm) of the form [38]

$$|I(f) - \widetilde{I}^K(f)| \leq \frac{U_\epsilon}{K^{\frac{1}{2}-\epsilon}}, \tag{25}$$

where $\epsilon \in (0, \frac{1}{2})$ is an arbitrarily small constant and U_ϵ is an a.s. finite random variable independent of K. The inequality (25) holds for every value of K, hence it is stronger than the classical CLT of Eq. (22). As Eq. (22), it displays the optimal Monte Carlo error rate $\mathcal{O}\left(\frac{1}{\sqrt{K}}\right)$, since $\epsilon > 0$ can be chosen as close to zero as desired.

B. Convergence of AIS estimators

The results summarized above hold for general importance samplers. In an AIS framework, however, it is of specific interest to study the convergence of the estimators as the proposals are adapted. This issue is tackled in the classical paper [40], where the estimators that result from aggregating weighted samples produced through several consecutive iterations are analyzed. Assuming that an AIS algorithm is run through J iterations, producing K samples per iteration for a total of JK samples overall (here we work with one proposal function per iteration), we construct the aggregated estimator of $I(f)$ as

$$\widetilde{I}^{J\times K}(f) = \frac{\sum_{j=1}^{J}\sum_{k=1}^{K} f(\mathbf{x}_j^{(k)}) w_j^{(k)}}{\sum_{j=1}^{J}\sum_{k=1}^{K} w_j^{(k)}}. \tag{26}$$

In the setup of [40], the proposal functions $q_j(\mathbf{x})$ are selected from a parametric family $q(\mathbf{x};\boldsymbol{\theta})$, where $\boldsymbol{\theta} = [\theta_1, \ldots, \theta_m]^\top \in \mathbb{R}^m$. The conditions to be satisfied by $q(\mathbf{x};\boldsymbol{\theta})$ are fairly general: $q(\mathbf{x};\boldsymbol{\theta})$ is a continuous function of $\boldsymbol{\theta}$, the weight function $w = \frac{\pi(\mathbf{x})}{q(\mathbf{x};\boldsymbol{\theta})}$ is uniformly bounded (over the space of $\mathbf{x}$ and $\boldsymbol{\theta}$), and $q(\mathbf{x};\boldsymbol{\theta}) > 0$ whenever $\pi(\mathbf{x}) > 0$. In addition, it is assumed that there exists an "optimal choice" of the proposal function, of the form $q(\mathbf{x};\boldsymbol{\theta}_o)$ where $\boldsymbol{\theta}_o = E_{\widetilde{\pi}}[\xi(\mathbf{x})]$ for some (possibly unknown) integrable function $\xi : \mathbb{R}^{d_x} \to \mathbb{R}^m$. The latter is a regularity assumption: it implies that, if the weights are proper and $K \to \infty$, it is possible to approximate the target

proposal $q(\mathbf{x};\boldsymbol{\theta}_o)$ as tightly as we wish. Under these assumptions, in [40] it is proved that

$$\lim_{J\times K\to\infty} \widetilde{I}^{J\times K}(f) = I(f) \quad \text{a.s., and} \quad \lim_{J\times K\to\infty} \sqrt{JK}(\widetilde{I}^{J\times K}(f) - I(f)) \overset{d}{=} \mathcal{N}(0,\sigma^2(f)), \quad (27)$$

where the limit variance $\sigma^2(f)$ is finite, and it depends on the test function and the normalization constant of $\widetilde{\pi}$. Convergence of the first limit in Eq. (27) guarantees consistency, while the second expression is a CLT that shows that the asymptotic optimal error rate $\mathcal{O}\left(\frac{1}{JK}\right)$ can be achieved without discarding any samples. Consistency of the aggregate estimator $\widetilde{I}^{J\times K}(f)$ can be proved in a rather straightforward manner for most AIS schemes as long as the importance weights are proper at each iteration and the weight function remains bounded, even if an "optimal" or "desired" proposal $q(\mathbf{x};\boldsymbol{\theta}_o)$ does not exist (or simply changes from one iteration to the next).

C. AIS and high-dimensional target pdfs

The error bounds of (24) and (25) or the variances in the CLT's (22) and (27) depend on the dimension d_x of the target random vector $\mathbf{X}$, often in an intricate manner. Few analytical results on the effect of the dimension are available in the literature. In simplified scenarios, and through numerical studies, it has been shown that often the number of samples K has to be increased exponentially with d_x in order to attain a prescribed performance [41]. However, it has *not* been proved that this is necessarily the case and some recent theoretical results actually suggest otherwise. In [42], the stability of the effective sample size (ESS), constructed as $\text{ESS}_j^K = \frac{\left(\sum_{k=1}^K w_j^{(k)}\right)^2}{\sum_{k=1}^K (w_j^{(k)})^2}$, of a sequential MC sampler as the dimension increases, $d_x \to \infty$, is analyzed. The ESS, related to the variance of the weights, is commonly used to assess the numerical stability of the adaptive algorithms and detect the degeneracy phenomenon. In this AIS scheme, the target pdf $\widetilde{\pi}(\mathbf{x})$ is approximated through a sequence of "bridge" densities $\pi_0(\mathbf{x}), \pi_1(\mathbf{x}), \ldots, \pi_j(\mathbf{x}), \ldots, \pi_J(\mathbf{x})$, where $\pi_0(\mathbf{x})$ is "sufficiently easy" to approximate via IS and $\pi_J(\mathbf{x}) = \widetilde{\pi}(\mathbf{x})$. The intuition is that we can start approximating π_0 and, assuming $\pi_{j-1}(\mathbf{x})$ and $\pi_j(\mathbf{x})$ are similar enough, we can then

move parsimoniously through the sequence of bridge pdf's until we obtain an approximation of $\widetilde{\pi}(\mathbf{x}) = \pi_J(\mathbf{x})$. In this setup, the proposal functions $q_j(\mathbf{x})$ are devised as Markov kernels that jump from $\pi_{j-1}(\mathbf{x})$ to $\pi_j(\mathbf{x})$. In the specific scheme analyzed in [42], the bridge pdfs are constructed by tempering, i.e., selecting a sequence of positive real numbers $0 < \epsilon_0 < \epsilon_1 < \cdots < \epsilon_J = 1$ and then setting $\pi_j(\mathbf{x}) = \widetilde{\pi}^{\epsilon_j}(\mathbf{x})$.

Under the strongly simplifying assumption of $\mathbf{X}$ being a vector of independent variables, i.e., $\widetilde{\pi}(\mathbf{x}) = \prod_{i=1}^{d_x} \widetilde{\pi}_i(x_i)$, but still assuming that the sample vector $\mathbf{x}_j^{(k)}$ is drawn jointly (and not independently, entry-wise) from the proposal $q_j(\mathbf{x})$, it is proved in [42] that $\lim_{d_x \to \infty} \mathrm{ESS}_j^K = C$ a.s., where C is a positive constant, even if the number of samples K is held constant. Moreover, this can be achieved when the number of bridge pdf's is $J = \mathcal{O}(d_x)$. These results indicate that this particular AIS method remains numerically stable (i.e., the weights do not degenerate) as the dimension d_x becomes arbitrarily large; however, they are mainly of theoretical (rather than practical) interest because of the strong assumptions involved. Nevertheless, they suggest that AIS schemes may beat the curse-of-dimensionality in some scenarios if properly designed.

D. A comparison of the convergence properties of IS and MCMC methods

MCMC [43] and AIS methods are often competing techniques to tackle the same class of inference problems, hence a brief comparison of their theoretical properties is relevant. MCMC schemes generate a chain of correlated samples $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(k)}, \ldots$ using a suitable Markov kernel $\mathcal{K}(\mathbf{x}^{(k-1)}, \mathbf{x}^{(k)})$ to draw $\mathbf{x}^{(k)}$ conditional on $\mathbf{x}^{(k-1)}$. Different algorithms, e.g., the Gibbs sampler or the Metropolis-Hastings (MH) method [43], yield different kernels. In any case, $\mathcal{K}(\cdot, \cdot)$ is designed so as to guarantee, under mild assumptions, that $\lim_{k \to \infty} p_k = \widetilde{\pi}$ a.s., where p_k denotes the pdf of the kth element of the chain, which generates $\mathbf{x}^{(k)}$, i.e., the generated sequence $\xi^{(k)}$, $k = 1, 2, ...$, has $\widetilde{\pi}$ as a stationary pdf [7], [43], [44]. There are no known rates for the convergence of p_k towards $\widetilde{\pi}$. However, it has been found that this rate can be very low in some scenarios.

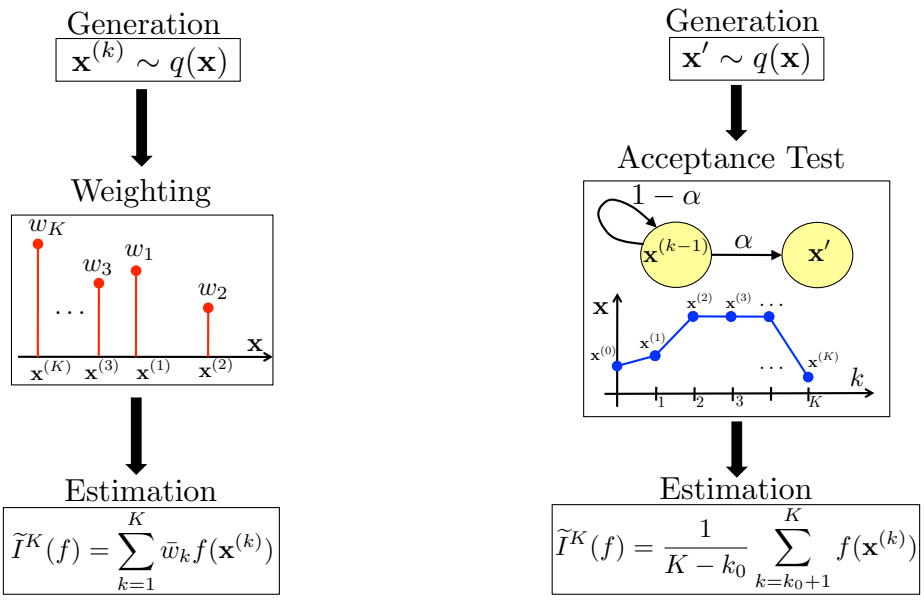
Moreover, it has to be taken into account that estimators constructed from an MCMC run of length K have the form

$$\widetilde{I}^K_{MCMC} \quad = \quad \frac{1}{K-k_0} \sum_{k=k_0+1}^{K} f(\mathbf{x}^{(k)}), \tag{28}$$

where the first k_0 samples are discarded to allow for the convergence of p_k. While $E[\widetilde{I}^K_{MCMC}(f)] \approx I(f)$, assuming $p_k \approx \widetilde{\pi}$, the random variates $f(\mathbf{x}^{(k)})$ are correlated and, therefore, the analysis of Var$(\widetilde{I}^K_{MCMC})$ is difficult. Again, it can be shown that $\widetilde{I}^K_{MCMC}(f) \to I(f)$ a.s. but no error rates are available.

This double asymptotics inherent to MCMC (we need the chain to *burn-in* so that $p_k \to \widetilde{\pi}$, then we need $K \to \infty$ for $\widetilde{I}^K_{MCMC}(f) \to I(f)$) often make these algorithms slower and computationally less efficient than AIS schemes [32], [38]. Moreover, in problems where the normalizing constant $Z = \left(\int \pi(\mathbf{x}) d\mathbf{x}\right)^{-1}$ is of interest (e.g., for model validation or model selection) AIS is a natural solution, as it readily yields unbiased estimates $\widehat{Z}^K_j = \frac{1}{K}\sum_{k=1}^K w^{(k)}$, $j = 1, ..., J$, while MCMC is often harder to apply [45]. There have been many recent attempts to devise algorithms that combine MCMC and AIS principles in order to take advantage of the strengths of both approaches [35], [46].

A pictorial comparison between IS and MCMC approaches is provided in Figure 5. In an MH-type sampler, a new state in the chain is proposed, and it is accepted or rejected with a suitable probability α. The number of repetitions of the same current state $\mathbf{x}^{(k)}$ plays the role of a weight in the estimator $\widetilde{I}^K_{MCMC}(f)$. However, unlike in IS, given a sample $\mathbf{x}^{(k)}$, the weighting procedure is not provided by a deterministic function (e.g., by $\frac{\pi(\mathbf{x})}{q(\mathbf{x})}$), but instead is a result of a stochastic process defined by the acceptance MCMC tests performed at each iteration.



(a) Importance Sampler (b) MH-type Sampler

Fig. 5: Graphical representation of importance sampling and MCMC procedures in order to provide an estimator $\widetilde{I}^K(f)$ of $I(f)$. More specifically, we have considered Metropolis-Hastings (MH) type of MCMC algorithms, where a novel possible state $\mathbf{x}'$ is drawn from $q(x)$, and it is accepted, thus setting $\mathbf{x}^{(k)} = \mathbf{x}'$ with a suitable probability α. Otherwise, the next state of the chain is set equal to the previous one, i.e., $\mathbf{x}^{(k)} = \mathbf{x}^{(k-1)}$ with probability $1 - \alpha$.

E. Parallelization

IS methods are easily parallelizable as the samples $\mathbf{x}^{(k)}$ are independent and, therefore, can be generated concurrently. In comparison, competing MCMC methods are much harder to parallelize, because the samples in a Markov chain are inherently sequential. With the availability of state-of-the-art multi-core computers and GPUs, this may be a key factor in favor of IS schemes. See [47] for a comparison of various MC schemes running on GPU systems.

In the specific case of AIS schemes, it is relatively straightforward to identify two stages in all of the presented algorithms. The first stage, that includes sampling and weighting, is a readily parallelizable task. This is the same as in standard IS, where each sample can (ideally) be generated and processed independently. The second stage, however, involves adaptation and, for some schemes, resampling. In this stage, it is necessary to process together all the samples and weights, e.g., to calculate the parameters of the new proposals in schemes like AMIS or N-PMC,

or even to run MCMC steps in the LAIS method. The adaptation step can be expected to be non-parallelizable, or parallelizable to a lesser extent, on standard computing devices.

F. Applications and challenges

While the range of applications of AIS algorithms is broad, it is worth discussing some particular fields where this methodology has either been applied with special success (compared to state-of-the-art techniques) or appears as a promising tool to tackle hard and long-standing problems.

The problems of detection and estimation in wireless sensor networks have been of great interest to the signal processing community for more than a decade. They involve scenarios where data related to a particular signal of interest are collected at various different sites of a network. Often, these observations can only be shared under tight constraints (due to scarce communication bandwidth, limited power, etc.) and estimation has to be performed with partial data or in a distributed fashion. One example of this class of problems, the localization of an object using signal-strength measurements, is presented in Section VI-A. A general challenge in this field is the design of schemes for the distributed implementation of AIS schemes with a minimal communication among the nodes of the network. Ideas based on the exchange of summary statistics have been explored, especially in the context of *sequential* importance sampling (see, e.g., [48]), but efficient schemes (accurate yet affordable in terms of both communication and computation) are still needed.

Another example explored in Section VI-B is the fitting of Gaussian processes (GPs) for nonlinear regression problems. GPs have found a plethora of applications in problems where one needs to approximate smooth functions for which a parametric model is not available at all, and the complete function has to be learned from a discrete collection of data-points [49]. While GPs are powerful models, their performance can be very sensitive to the fitting of a number of hyperparameters. The example in Section VI-B shows that AIS can efficiently tackle this problem.

AIS has also shown advantage compared to state-of-the-art methods in performing inference for stochastic kinetic models (SKMs) [32]. SKMs are used in biochemistry or ecology to model complex interactions among populations of different species [50]. In ecology, SKMs yield a generalization of classical predator-prey models. In biochemistry, an SKM represents a system with n types of molecules (species) and k types of reactions. In both cases, it is of interest to track and predict the species populations, which evolve as a multidimensional continuous-time jump process, and estimate the rates that govern the dynamics. It has been shown [32] that AIS schemes (in this case, the NPMC algorithm) can attain the same performance as state-of-the-art particle MCMC methods [51] with a fraction of the computational cost for modest SKMs. The accurate fitting of complex, high-dimensional SKMs is an open problem with outstanding real-world applications.

AIS techniques also enable consistent parameter estimation in α-stable distributions with very heavy tails [38]. α stable distributions are often denoted as $S(\alpha, \beta, \gamma, \delta)$, where $0 < \alpha \leq 2$ determines the weight of the tails (the smaller the value of α, the heavier the tails), β is a skewness parameter, and $\gamma > 0$ and δ determine the scale and location. Except for particular cases, the associated pdf's can only be approximated numerically. Fast, classical methods for parameter estimation are known to work only for $\alpha \geq 0.5$ (i.e., with moderate tails). The results in [38], including an example with real data, show that AIS methods can overcome this limitation and open the door to address problems formerly intractable.

Finally, a challenging arena for the application of AIS methods includes a number of problems where very large scale models are used and need to be fitted from (often scarce) data. This includes many large-scale systems used in geophysics, for example in oceanography [52], climate modeling [53] or cosmology [54]. In all these cases, algorithms that attain a good trade-off between computational complexity and accuracy of the resulting estimators are very much needed and advanced AIS holds potential to be successfully applied.

VI. Numerical Examples

A. Localization problem in a wireless sensor network

We consider the problem of positioning a target in a wireless sensor network using range measurements [55]. We assume that the measurements of the sensors are contaminated by additive white Gaussian noise (AWGN) with different unknown powers. This situation is common in many practical scenarios where, even if the sensors are of the same manufacturer and model, the noise level can be different due to various factors. They include signal propagation conditions, manufacturing imperfections, and environmental conditions (e.g., humidity or temperature). Moreover, these conditions can change over time. Hence, in practice the central node of the network has to re-estimate the noise powers (in addition to the target's position and possibly other parameters of the model) whenever a new block of observations is acquired.

More specifically, we denote the unknown target's position with the random vector $\mathbf{\Lambda} = [\Lambda_1, \Lambda_2]^\top$, and a specific realization of it as $\boldsymbol{\lambda}$. Let there be M sensors at locations $\mathbf{h}_m$, $m = 1, 2, \cdots, M$. The model for the observations is

$$y_{i,m} = 20 \log \left(||\boldsymbol{\lambda} - \mathbf{h}_m|| \right) + v_{i,m}, \quad m = 1, \ldots, M; \quad i = 1, 2, \ldots, N_o \tag{29}$$

where $\| \cdot \|$ denotes the L_2 norm, $y_{i,m}$ is the ith observation of the Mth sensor, N_o is the number of observations of each of the sensors, and the $v_{i,m}$'s are independent Gaussian random variables with pdfs $\mathcal{N}(v_{i,m}; 0, \gamma_m^2)$, $m = 1, \ldots, M$. We denote the vector of standard deviations as $\boldsymbol{\gamma} = [\gamma_1, \ldots, \gamma_M]$. We adopt a uniform prior $\mathcal{U}(\mathcal{R}_\lambda)$ for the position $[\Lambda_1, \Lambda_2]^\top$, over a predefined support, and a uniform prior for γ_j, also over a preset range, $\mathcal{R}_\gamma$. Thus, the posterior pdf is

$$\begin{aligned} \widetilde{\pi}(\boldsymbol{\lambda}, \boldsymbol{\gamma} | \mathbf{Y}) &\propto \ell(\mathbf{y} | \lambda_1, \lambda_2, \gamma_1, \ldots, \gamma_M) \prod_{i=1}^{2} p(\lambda_i) \prod_{m=1}^{M} p(\gamma_m), \\ &= \left[\prod_{i=1}^{N_o} \prod_{m=1}^{M} \frac{1}{\sqrt{2\pi\gamma_m^2}} \exp\left(-\frac{1}{2\gamma_m^2} (y_{i,m} - 20 \log \left(||\boldsymbol{\lambda} - \mathbf{h}_m|| \right)^2 \right) \right] \mathbb{I}(\mathcal{R}_\lambda) \mathbb{I}(\mathcal{R}_\gamma), \end{aligned} \tag{30}$$

where N_o is the number of observations, $y_{i,m}$ is the ith observation of the mth sensor, and $\mathbb{I}_c(\mathcal{S})$ is an indicator function that takes a value equal to one if $c \in \mathcal{S}$, and is equal to zero otherwise. Thus, in this problem $\mathbf{x} = [\boldsymbol{\lambda}^\top, \boldsymbol{\gamma}^\top]^\top$, and $d_x = M + 2$.

Our goal is to compute the Minimum Mean Squared Error (MMSE) estimate, which corresponds to the expected value of the posterior $\widetilde{\pi}(\boldsymbol{\lambda}, \boldsymbol{\gamma}|\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_M)$, where the $\mathbf{y}_m$s are vectors whose elements are the measurements of the mth sensor. Since the MMSE estimate cannot be computed analytically, we applied several AIS methods to approximate it via MC quadrature. In particular, we worked with the standard PMC method [19], two different DM-PMC techniques [24], AMIS [21] and LAIS [23].

In our experiment, we had $M = 6$ sensors, and the locations of the sensors were at $\mathbf{h}_1 = [3, -8]^\top$, $\mathbf{h}_2 = [8, 10]^\top$, $\mathbf{h}_3 = [-4, -6]^\top$, $\mathbf{h}_4 = [-8, 1]^\top$, $\mathbf{h}_5 = [10, 0]^\top$ and $\mathbf{h}_6 = [0, 10]^\top$. In all the cases, we employed Gaussian proposal densities, $q_{n,j}(\mathbf{x}|\boldsymbol{\mu}_{n,j}, \mathbf{C}_{n,j}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_{n,j}, \mathbf{C}_{n,j})$ with $\boldsymbol{\mu}_{n,1} \sim \mathcal{U}([1, 4]^{d_x})$ for $n = 1, \ldots, N$. The target was located at $\boldsymbol{\lambda} = [\lambda_1 = 2.5, \lambda_2 = 2.5]^\top$, and the vector of standard deviations was $\boldsymbol{\gamma} = [\gamma_1 = 1, \gamma_2 = 2, \gamma_3 = 1, \gamma_4 = 0.5, \gamma_5 = 3, \gamma_6 = 0.2]$. We generated $N_o = 20$ observations for each sensor according to the model given by Eq. (29). The uniform prior $\mathcal{U}(\mathcal{R}_\lambda)$ over the position $[\lambda_1, \lambda_2]^\top$ had a support $\mathcal{R}_\lambda = [-30 \times 30]^2$, and the uniform prior of the γ_is was $\mathcal{U}([0.01, 20])$. Thus, the overall prior of $\boldsymbol{\gamma}$ was $\mathcal{U}(\mathcal{R}_{\boldsymbol{\gamma}})$ with $\mathcal{R}_{\boldsymbol{\gamma}} = [0.01, 20]^M$. Then, we obtained the measurement vectors $\mathbf{y}_1, \ldots, \mathbf{y}_M$, where $\mathbf{y}_i \in \mathbb{R}^{N_o}$. Note that, regarding the dimension of the observations, we have $d_y = N_o M = 120$.

For the PMC, the DM-PMCs and LAIS we set $\mathbf{C}_{n,j} = \mathbf{C}_n = \mathbf{C} = \sigma^2\mathbf{I}$ with $\sigma = 1$. In AMIS, we have $N = 1$ and $\mathbf{C}_{n,j} = \mathbf{C}_j = \sigma_j^2\mathbf{I}$, and we set $\sigma_1 \in \{1, 2\}$. In the adaptation layer of LAIS, in order to obtain $\{\mu_{n,j}\}_{n=1}^N$ from the previous population $\{\mu_{n,j-1}\}_{n=1}^N$, we employ parallel Metropolis-Hastings chains with a Gaussian random-walk proposal pdf, $\varphi_n(\boldsymbol{\mu}_{n,j}|\boldsymbol{\mu}_{n,j-1}, \sigma^2\mathbf{I}) = \mathcal{N}(\boldsymbol{\mu}_{n,j}|\boldsymbol{\mu}_{n,j-1}, \sigma^2\mathbf{I})$ with $\sigma = 1$. Moreover, we also test the application of N independent parallel Metropolis-Hastings algorithms with the same Gaussian random-walk proposal pdf, $\varphi_n(\boldsymbol{\mu}_{n,t}|\boldsymbol{\mu}_{n,j-1}, \sigma^2\mathbf{I})$,

employed in the adaptation of LAIS.

We fix the total number of evaluations of the posterior density to $E = 10^4$, since this is usually the most costly step in MC algorithms. Let us recall that J denotes the total number of iterations and K the number of samples drawn from each proposal at each iteration. Moreover, we denote as S the total number of samples employed in the final IS estimator. In LAIS, the total number of evaluations of the target pdf is $E = NJ(K+1)$, whereas $S = NJK$ (i.e., $E > S$ due to the use of the Markov adaptation process). For the rest of the methods, we have $E = S = NKJ$ (note that $N = 1$ in AMIS, while $K = 1$ in standard PMC and Metropolis-Hastings). Several combinations of N, J and K are tested for the fixed $E = 10^4$ evaluations.

We computed the Mean Squared Error (MSE) of the different estimators obtained w.r.t. the ground-truth, $\mathbf{x} = [\boldsymbol{\lambda}^\top, \boldsymbol{\gamma}^\top]^\top$. The results, averaged over 500 independent runs, are provided in Tables VII–XII (one table per technique) with the best and worst MSE values highlighted in boldface. In this particular experiment, with a unimodal posterior pdf and a good initialization $\boldsymbol{\mu}_{n,1} \sim \mathcal{U}([1,5]^{d_x})$, the PMC techniques and the AMIS method provide the smallest MSE values. The standard PMC method seems to perform better if one uses a larger value of N and a smaller number of iterations J. In fact, the use of a small number of proposal pdfs can lead to catastrophic results in this case. The DM-PMC techniques substantially mitigate this problem, with GR-DM-PMC showing a more robust behavior w.r.t. the parameter choice than LR-DM-PMC (note that GR stands for global resampling and LR for local resampling). AMIS provides very good results, although it shows some sensitivity w.r.t. the choice of the initial scale parameter, σ_1. Note that LAIS provides slightly worse results than AMIS, but also shows less sensitivity w.r.t. the parameter choice and outperforms the performance of N independent parallel MH chains. Finally, Fig. 6 shows the evolution of the estimators of AMIS ($J = 300$, $K = 200$) and standard PMC ($N = 1000$, $J = 100$) as functions of the number of iterations, j, in one specific run.

TABLE VII: **Results standard PMC [19] (localization example).**

MSE	**25.12**	3.96	1.35	1.08	0.72	**0.61**	0.70
N	5	10	50	100	500	1000	2000
J	2000	1000	200	100	20	10	5
E	$S = NJ = 10^4$						
Range	**Min MSE = 0.61 ——— Max MSE = 25.12**						

TABLE VIII: **Results GR-DM-PMC [24] (localization example).**

MSE	0.96	0.89	**0.75**	0.84	0.85	**1.47**	0.81	0.76	0.79	0.84	0.80	0.81
N	5	5	5	10	10	10	50	50	100	100	500	1000
J	50	100	10	10	5	200	5	10	5	10	5	5
K	40	20	200	100	200	5	40	20	20	10	4	2
E	$S = NTM = 10^4$											
Range	**Min MSE = 0.75 ——— Max MSE = 1.47**											

TABLE IX: **Results LR-DM-PMC [24] (localization example).**

MSE	1.14	1.52	**0.77**	**0.77**	0.79	**2.91**	1.01	1.24	1.26	1.44	1.32	1.49
N	5	5	5	10	10	10	50	50	100	100	500	1000
J	50	100	10	10	5	200	5	10	5	10	5	5
K	40	20	200	100	200	5	40	20	20	10	4	2
E	$S = NTM = 10^4$											
Range	**Min MSE = 0.77 ——— Max MSE = 2.91**											

TABLE X: **Results AMIS [21] (localization example).**

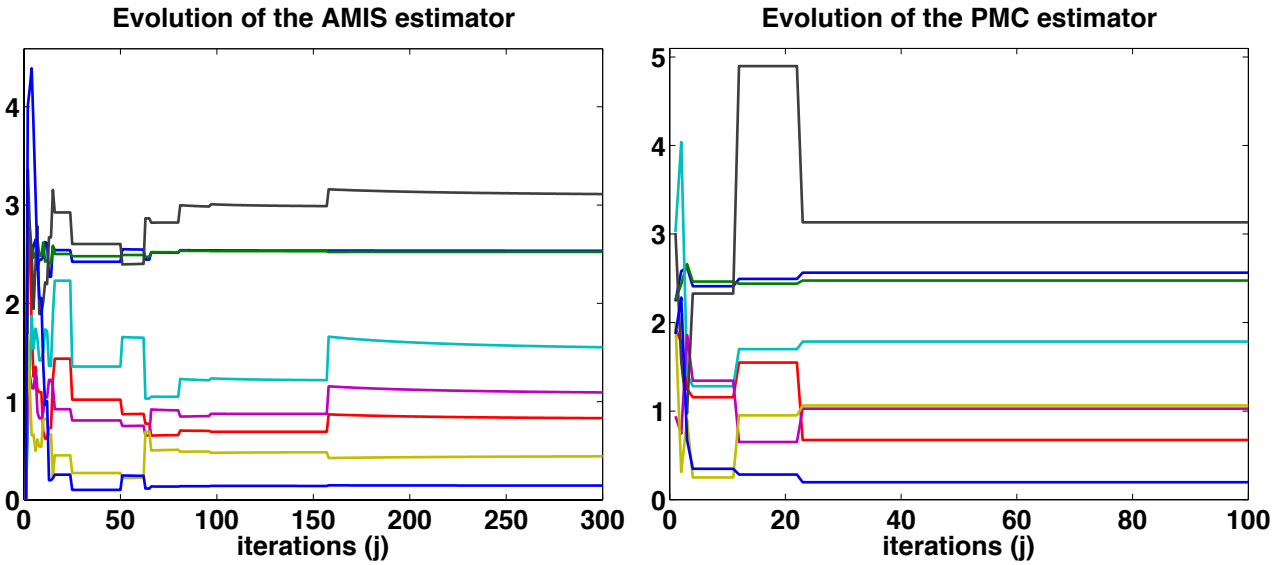
MSE ($\sigma_0 = 1$)	0.80	**0.72**	0.75	0.76	0.88	1.29
MSE ($\sigma_0 = 2$)	1.53	1.48	1.42	1.29	1.48	**1.71**
N	1					
J	200	100	50	20	10	5
K	50	100	200	500	1000	2000
E	$S = TM = 10^4$					
Range	**Min MSE = 0.72 ——— Max MSE = 1.71**					

TABLE XI: **Results LAIS [23] (localization example).**

MSE	**1.91**	1.52	1.14	1.11	1.10	**1.06**	1.29	1.25	1.26	1.30	1.41
N	1	2	5	5	10	10	100	100	100	200	10^3
J	$5 \cdot 10^3$	500	250	500	250	500	10	25	50	25	5
K	1	9	7	3	3	1	9	3	1	1	1
S	$5 \cdot 10^3$	$9 \cdot 10^3$	8750	7500	7500	$5 \cdot 10^3$	$9 \cdot 10^3$	7500	$5 \cdot 10^3$	$5 \cdot 10^3$	$5 \cdot 10^3$
E	$S + NT = NT(M+1) = 10^4$										
Range	**Min MSE = 1.06 ——— Max MSE = 1.91**										

TABLE XII: **Results independent Metropolis-Hastings parallel chains (localization example).**

MSE	1.42	**1.31**	1.44	2.32	2.73	**3.21**	3.18	3.15
N	1	5	10	50	100	500	1000	2000
J	10^4	$2 \cdot 10^3$	10^3	200	100	20	10	5
E	$S = NT = 10^4$							
MSE range	**Min MSE = 1.31** ——— **Max MSE = 3.21**							



(a) (b)

Fig. 6: Evolution of AMIS ($T = 300$, $M = 200$) and standard PMC ($N = 1000$, $T = 100$) estimators as functions of the number of iterations, j, in one specific run.

B. Learning hyperparameters for Gaussian process regression models

Gaussian processes (GPs) are a modern machine learning approach to solving regression problems [56]. Given a covariance kernel function, learning its hyper-parameters is the key to attain accurate performance. In this section, we test the different AIS schemes for estimating the hyper-parameters of a Gaussian process (GP) regression model.

Let us assume that we have a set of observed data pairs, $\{y_i, \mathbf{z}_i\}_{i=1}^P$ with $y_i \in \mathbb{R}$ and $\mathbf{z}_i \in \mathbb{R}^L$, and let us denote the corresponding $P \times 1$ output vector as $\mathbf{y} = [y_1, \ldots, y_P]^\top$ and the $L \times P$ input matrix as $\mathbf{Z} = [\mathbf{z}_1, \ldots, \mathbf{z}_L]$. We address the problem of inferring the unknown function f that links the variables y and $\mathbf{z}$. Namely, the assumed model is $y = f(\mathbf{z}) + e$, where $e \sim N(e; 0, \sigma^2)$ and $f(\mathbf{z})$ is a realization of a GP, $f(\mathbf{z}) \sim \mathcal{GP}(\mu(\mathbf{z}), \kappa(\mathbf{z}, \mathbf{r}))$ with $\mathbf{z}, \mathbf{r} \in \mathbb{R}^L$, $\mu(\mathbf{z}) = 0$, and the

kernel function has the form

$$\kappa(\mathbf{z},\mathbf{r}) = \exp\left(-\sum_{\ell=1}^{L}\frac{(z_\ell - r_\ell)^2}{2\alpha^2}\right). \tag{31}$$

(We point out that $f(\cdot)$ in this section has nothing to do with the test function used earlier in the paper.) Given these assumptions, the vector $\mathbf{f} = [f(\mathbf{z}_1),\ldots,f(\mathbf{z}_P)]^\top$ is distributed as $p(\mathbf{f}|\mathbf{Z},\alpha,\kappa) = \mathcal{N}(\mathbf{f};\mathbf{0},\mathbf{K})$, where $\mathbf{0}$ is a $P\times 1$ null vector, and $[\mathbf{K}]_{ij} := \kappa(\mathbf{z}_i,\mathbf{z}_j)$ for all $i,j = 1,\ldots,P$, is a $P\times P$ matrix. Therefore, $d_x = 2$ and the vector containing the hyper-parameters of the model is $\mathbf{x} = [x_1 = \alpha, x_2 = \sigma] \in \mathbb{R}^2$, where α is the hyper-parameter of the kernel function in Eq. (31), and σ is the standard deviation of the observation noise. In this experiment, we focus on the marginal posterior density of the hyperparameters [56], $\widetilde{\pi}(\mathbf{x}|\mathbf{y},\mathbf{Z},\kappa) \propto p(\mathbf{y}|\mathbf{x},\mathbf{Z},\kappa)p(\mathbf{x})$, which can be evaluated analytically, but we cannot compute integrals involving it. Considering a uniform prior $p(\mathbf{x})$ over $[0.01,20]^2$, and since $p(\mathbf{y}|\mathbf{x},\mathbf{Z},\kappa) = \mathcal{N}(\mathbf{y};\mathbf{0},\mathbf{K}+\sigma^2\mathbf{I})$, we have

$$\log\left[\pi(\mathbf{x}|\mathbf{y},\mathbf{Z},\kappa)\right] = -\frac{1}{2}\mathbf{y}^\top(\mathbf{K}+\sigma^2\mathbf{I})^{-1}\mathbf{y} - \frac{1}{2}\log\left[\det\left(\mathbf{K}+\sigma^2\mathbf{I}\right)\right], \tag{32}$$

where $\mathbf{K}$ depends on α [56]. Since the moments of this marginal posterior cannot be computed analytically, we use again MC integration with different AIS methods to approximate the MMSE estimator, $\widehat{\mathbf{x}} = [\widehat{\alpha},\widehat{\sigma}]$, which corresponds to the expected value of $\mathbf{X}$ w.r.t. $\widetilde{\pi}(\mathbf{x}|\mathbf{y},\mathbf{Z},\kappa)$.

For this experiment, we generated $P = 200$ pairs of data, $\{y_j,\mathbf{z}_j\}_{j=1}^P$, according to the previous GP model with $\alpha = 3$, $\sigma = 10$, $L = 1$ and $z_j \sim \mathcal{U}([0,10])$. Fixing the generated data, we then computed the true value of the MMSE, $\widehat{\mathbf{x}} = [\widehat{\alpha},\widehat{\sigma}] \approx [3.5200, 9.2811]$, using an exhaustive and costly grid search approximation, in order to compare the different AIS techniques. The corresponding posterior pdf is given in Fig. 7(a).

We compared the standard PMC method [19], the LR-DM-PMC technique [24], the AMIS [21] and the LAIS [23] algorithms. Again, for all of them we considered Gaussian proposal densities, $q_{n,j}(\mathbf{x}|\boldsymbol{\mu}_{n,j},\mathbf{C}_{n,j}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_{n,j},\mathbf{C}_{n,j})$ with $\boldsymbol{\mu}_{n,1} \sim \mathcal{U}([1,4]^2)$ for $n = 1,\ldots,N$. Note that, unlike

in the previous experiment, the true value of $\mathbf{x}$ does not belong to the initialization region $[1,4]^2$. For PMC, LR-DM-PMC, and LAIS we set $\mathbf{C}_{n,j} = \mathbf{C}_n = \mathbf{C} = \sigma^2\mathbf{I}$ with $\sigma = 2$. For AMIS, we had $N = 1$ and $\mathbf{C}_{n,j} = \mathbf{C}_j = \sigma_j^2\mathbf{I}$ and we set $\sigma_1 = 2$. In the adaptation layer of LAIS, in order to obtain $\{\mu_{n,j}\}_{n=1}^N$ from the previous population $\{\mu_{n,j-1}\}_{n=1}^N$, we employed parallel MH chains with a Gaussian random-walk proposal pdf, $\varphi_n(\boldsymbol{\mu}_{n,j}|\boldsymbol{\mu}_{n,j-1}, \sigma^2\mathbf{I}) = \mathcal{N}(\boldsymbol{\mu}_{n,j}|\boldsymbol{\mu}_{n,j-1}, \sigma^2\mathbf{I})$ with $\sigma = 2$. Once more, we fixed the total number of evaluations of the posterior pdf to $E = 10^4$ and we tested the algorithms considering different combinations of the parameters.

The results, in terms of MSE in the estimation of $\mathbf{x}$, are given in Tables XIII, XIV, XV and XVI. They were averaged over 500 independent runs. In this numerical experiment, LAIS and LR-DM-PMC provided smaller MSEs. The reason is that they discover and explore faster the tail of the posterior distribution with respect to the other techniques. The adaptation of the location parameters produced in one specific run by LAIS ($N = 5$ and $T = 100$) is shown in Fig. 7(b).

TABLE XIII: **Results standard PMC [19] (GP example).**

MSE	**0.44**	0.87	1.01	0.88	0.86	0.95	**1.15**
N	5	50	100	200	500	1000	2000
T	2000	200	100	50	20	10	5
E	$S = NT = 10^4$						
Range	**Min MSE = 0.44** ——— **Max MSE = 1.15**						

TABLE XIV: **Results LR-DM-PMC [24] (GP example).**

MSE	0.41	0.39	0.16	0.09	**0.04**	0.23	0.07	**0.46**
N	5	5	5	50	50	100	100	1000
T	10	20	40	10	20	10	20	5
M	200	100	50	20	10	10	5	2
E	$S = NTM = 10^4$							
Range	**Min MSE = 0.04** ——— **Max MSE = 0.46**							

VII. Concluding remarks and outlook

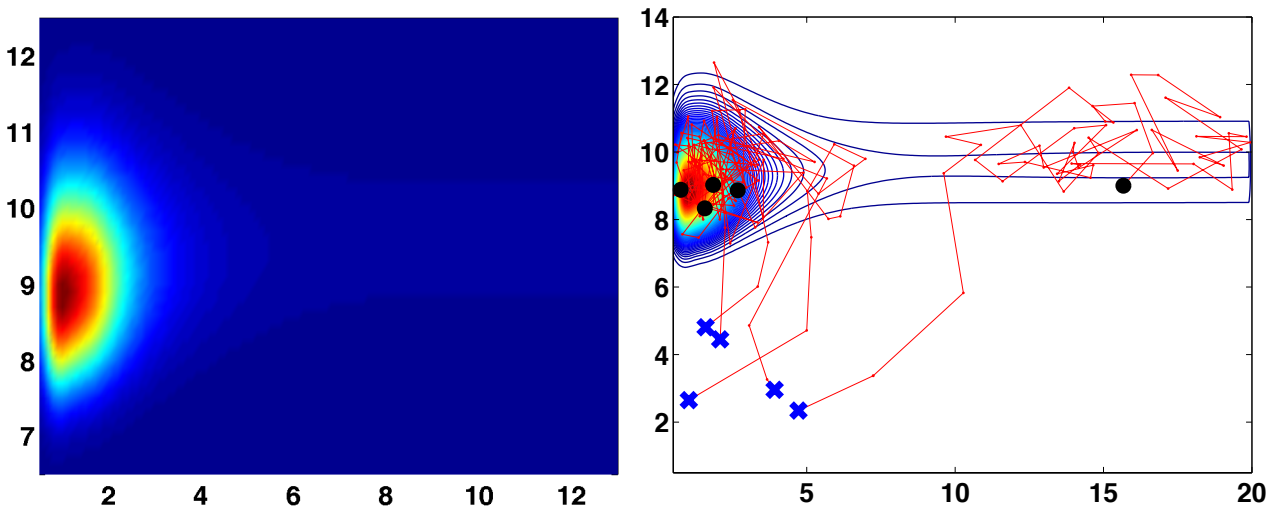
In signal processing, an important task is making inference from data about model parameters or models in general. From a Bayesian point of view, ideally this inference is made from posterior

TABLE XV: **Results AMIS [21] (GP example).**

MSE	1.32	**1.35**	1.26	1.27	**1.23**
N	1				
T	200	100	50	20	10
M	50	100	200	500	1000
E	$S = TM = 10^4$				
Range	**Min MSE = 1.23** ——— **Max MSE = 1.35**				

TABLE XVI: **Results LAIS [23] (GP example).**

MSE	**1.04**	0.46	0.21	0.09	**0.03**	0.31	0.65
N	1	5	10	50	100	500	1000
T	5000	1000	500	100	50	10	5
M	1						
E	$NT(M+1) = 10^4$						
Range	**Min MSE = 0.03** ——— **Max MSE = 1.04**						



(a) (b)

Fig. 7: **(a)** Posterior density $\pi(\mathbf{x}|\mathbf{y}, \mathbf{Z}, \kappa)$. **(b)** Evolution of the location parameters $\boldsymbol{\mu}_{n,t}$ in one specific run of LAIS with $N = 5$ and $T = 100$ (jointly with the contour plot of the posterior pdf). The starting points are shown with x-marks whereas the final locations are depicted with circles.

distributions of the unknowns. For complex models, it is very difficult to find these posteriors. In such cases, one resorts to approximations in the sense that one generates samples that are drawn from the posterior distributions. A tool that helps practitioners to get such samples is MCMC sampling. As already pointed out, the MCMC algorithms and the growth of computing power have invigorated the Bayesian methodology in the last two and a half decades to the point that

today we use them to solve most intricate problems.

In this article, we have argued that practitioners of signal processing should be aware of another option for solving inference problems by way of drawing samples from distributions. It is based on a methodology known as AIS. AIS methods have the subtle ability to learn the pdfs that produce better samples for constructing posteriors and that eventually allow for a more accurate inference. The learning is accomplished in iterations where the samples from previous iterations serve to find better proposal pdfs.

AIS is often simpler to implement than MCMC sampling. Besides simplicity, AIS has other advantages over MCMC sampling, including that it does not produce correlated samples, that there is no such thing as burn-in period, and that AIS is easier for parallelization. We also have better understanding of the rates of convergence of AIS methods than those of MCMC sampling. A pitfall of importance sampling methods is the possibility of using proposal pdfs with thinner tails than those of the target distribution which can easily ruin any estimate from the generated data and the computed weights.

In this article, we have surveyed the state-of-the-art of AIS methods and the advances in the area in recent years. Next we provide some direction of future work.

The most important open problem of AIS, as we have already alluded, is the development of AIS methods that can work accurately in high dimensional spaces. As the dimension of $\mathbf{x}$ increases, the complexity of finding good proposal pdfs explodes (curse-of-dimensionality). One approach for resolving this problem is to work with compartmentalized spaces of the unknowns and accept that we will not have approximations of the full joint posterior but instead, a number of marginalized posteriors.

Another way of addressing high-dimensionality is by particle flows. This approach has been of interest in particle filtering, where samples drawn from the prior distribution are migrated to the posterior distribution of the unknowns by solving partial differential equations [57]. Even though

the problems we solve with AIS are different from those addressed by particle filtering, there is enough common ground between the two methodologies to investigate the application of particle flows to AIS. How can the underlying principles of particle flows be exploited in AIS?

In recent years, stochastic optimization methods have seen a resurgence. One reason for this is that there are many problems that can be formulated as optimization problems where the minimized objective function is a sum of many loss functions. Importance sampling is one of a number of Monte Carlo sampling-based methods for stochastic optimization. It can improve the convergence rate of the optimization and reduce the stochastic variance of the result [58]. The use of AIS for optimization raises various challenging questions, including convergence to optimal solutions and optimal values.

A specific application of stochastic optimization is in stochastic variational Bayesian methods. These methods can be applied to complex probabilistic models and large data sets with a vast range of applications in machine learning. Recently, a synthesis between variational inference and MCMC sampling for variational approximation has been proposed [59]. It was claimed that a fast posterior approximation through the maximization of an explicit objective was accomplished. Furthermore, the proposed method offered trade offs between computation and accuracy. Clearly, AIS is a natural candidate to be applied in the same setting with the possibility of performing even better than MCMC sampling.

Finally, in the years to come, we expect that AIS methods will find increased use within the signal processing community. Much of the research in this area will be driven by novel applications and by models with expanded complexity. There will be new applications that may even include use of AIS in deep learning for computing the weights of the hidden layers. The addressed problems will not only require estimating unknown quantities, but also finding the best models from a set of predefined models or finding the best model in nonparametric Bayesian settings where the number of models is not set a priori.

Acknowledgments

The authors gratefully acknowledge the support for their research, the National Science Foundation under Awards CCF-1617986 (M. F. Bugallo) and CCF-1618999 (P. M. Djurić); Ministerio de Economía y Competitividad of Spain under TEC2015-69868-C2-1-R ADVENTURE and the Office of Naval Research Global under Award N62909-15-1-2011 (J. Míguez); the European Research Council (ERC) through the ERC Consolidator Grant SEDAL ERC-2014-CoG 647423 (L. Martino); Ministerio de Economía y Competitividad of Spain under TEC2015-64835-C3-3-R MIMOD-PLC project, Ministerio de Educación, Cultura y Deporte of Spain under CAS15/00350 grant, and Universidad Politécnica de Madrid through a mobility grant for a short visit to Stony Brook University (D. Luengo).

REFERENCES

[1] X. Sheng and Y.-H. Hu, "Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks," *IEEE Transactions on Signal Processing*, vol. 53, no. 1, pp. 44–53, 2005.

[2] Z. Chen, F. Xia, T. Huang, F. Bu, and H. Wang, "A localization method for the Internet of Things," *The Journal of Supercomputing*, pp. 1–18, 2013.

[3] C. Phillips, J. Mattout, M. D. Rugg, P. Maquet, and K. J. Friston, "An empirical Bayesian solution to the source reconstruction problem in eeg," *NeuroImage*, vol. 24, no. 4, pp. 997–1011, 2005.

[4] A. Kuklasiński, S. Doclo, S. H. Jensen, and J. Jensen, "Maximum likelihood PSD estimation for speech enhancement in reverberation and noise," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 24, no. 9, pp. 1595–1608, 2016.

[5] I. Shmulevich and E. R. Dougherty, *Genomic Signal Processing*, Princeton University Press, 2014.

[6] N. Metropolis, "The beginning of the Monte Carlo method," *Los Alamos Science*, pp. 125–130, 1987.

[7] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*, Springer, 2004.

[8] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953.

[9] W. K. Hastings, "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika*, vol. 57, no. 1, pp. 97–109, 1970.

[10] W. R. Gilks, S. Richardson, and D. Spiegelhalter, *Markov chain Monte Carlo in Practice*, CRC press, 1995.

[11] H. Kahn, "Random sampling (Monte Carlo) techniques in neutron attenuation problems.," *Nucleonics*, vol. 6, no. 5, pp. 27–passim, 1950.

[12] T. Hesterberg, "Weighted average importance sampling and defensive mixture distributions," *Technometrics*, vol. 37, pp. 185–194, 1995.

[13] J. M. Bernardo and A. F. M. Smith, *Bayesian Theory*, Wiley & sons, 1994.

[14] N. Metropolis and S. Ulam, "The Monte Carlo method," *Journal of the American Statistical Association*, vol. 44, pp. 335–341, 1949.

[15] G. Rubino and B. Tuffin, *Rare Event Simulation using Monte Carlo Methods*, John Wiley & Sons, 2009.

[16] A. Doucet and A. M. Johansen, "A tutorial on particle filtering and smoothing: Fifteen years later," *Handbook of nonlinear filtering*, vol. 12, no. 656-704, pp. 3, 2009.

[17] E. Veach and L. Guibas, "Optimally combining sampling techniques for Monte Carlo rendering," in *Proceedings of the Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 1995.

[18] A. Owen and Y. Zhou, "Safe and effective importance sampling," *Journal of the American Statistical Association*, vol. 95, no. 449, pp. 135–143, 2000.

[19] O. Cappé, A. Guillin, J.-M. Marin, and C. P. Robert, "Population Monte Carlo," *Journal of Computational and Graphical Statistics*, vol. 13, no. 4, pp. 907–929, 2004.

[20] O. Cappé, R. Douc, A. Guillin, J. M. Marin, and C. P. Robert, "Adaptive importance sampling in general mixture classes," *Statistics and Computing*, vol. 18, pp. 447–459, 2008.

[21] J. M. Cornuet, J.-M. Marin, A. Mira, and C. P. Robert, "Adaptive multiple importance sampling," *Scandinavian Journal of Statistics*, vol. 39, no. 4, pp. 798–812, December 2012.

[22] L. Martino, V. Elvira, D. Luengo, and J. Corander, "An adaptive population importance sampler: Learning from the uncertanity," *IEEE Transactions on Signal Processing*, vol. 63, no. 16, pp. 4422–4437, 2015.

[23] L. Martino, V. Elvira, D. Luengo, and J. Corander, "Layered adaptive importance sampling," *Statistics and Computing*, vol. 27, no. 3, pp. 599–623, 2017.

[24] V. Elvira, L. Martino, D. Luengo, and M. F. Bugallo, "Improving Population Monte Carlo: Alternative weighting and resampling schemes," *Signal Processing*, vol. 131, no. 12, pp. 77–91, 2017.

[25] V. Elvira, L. Martino, D. Luengo, and M. F. Bugallo, "Generalized multiple importance sampling," *preprint: arXiv:1511.03095*, 2015.

[26] V. Elvira, L. Martino, D. Luengo, and M. F. Bugallo, "Efficient multiple importance sampling estimators," *Signal Processing Letters, IEEE*, vol. 22, no. 10, pp. 1757–1761, 2015.

[27] V. Elvira, L. Martino, D. Luengo, and M. F. Bugallo, "Heretical multiple importance sampling," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1474–1478, 2016.

[28] M. F. Bugallo, L. Martino, and J. Corander, "Adaptive importance sampling in signal processing," *Digital Signal Processing*, vol. 47, pp. 36–49, 2015.

[29] T. Li, M. Bolic, and P. M. Djurić, "Resampling methods for particle filtering: Classification, implementation, and strategies," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 70–86, 2015.

[30] G. R. Douc, J.-M. Marin, and C. Robert, "Convergence of adaptive mixtures of importance sampling schemes," *Annals of Statistics*, vol. 35, pp. 420–448, 2007.

[31] G. R. Douc, J.-M. Marin, and C. Robert, "Minimum variance importance sampling via population Monte Carlo," *ESAIM: Probability and Statistics*, vol. 11, pp. 427–447, 2007.

[32] E Koblents and J. Míguez, "A population monte carlo scheme with transformed weights and its application to stochastic kinetic models," *Statistics and Computing*, vol. 25, no. 2, pp. 407–425, 2015.

[33] R. J. Steele, A. E. Raftery, and M. J. Emond, "Computing normalizing constants for finite mixture models via incremental mixture importance sampling (IMIS)," *Journal of Computational and Graphical Statistics*, vol. 15, pp. 712–734, 1996.

[34] V. Elvira, L. Martino, D. Luengo, and J. Corander, "A gradient adaptive population importance sampler," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 4075–4079.

[35] Z. I. Botev, P. L'Ecuyer, and B. Tuffin, "Markov chain importance sampling with applications to rare event probability estimation," *Statistics and Computing*, vol. 23, no. 2, pp. 271–285, 2013.

[36] P. Del Moral, *Feynman-Kac Formulae: Genealogical and Interacting Particle Systems with Applications*, Springer, 2004.

[37] J. Geweke, "Bayesian inference in econometric models using Monte Carlo integration," *Econometrica*, vol. 24, pp. 1317–1399, 1989.

[38] E. Koblents, J. Miguez, M. A. Rodriguez, and A. M. Schmidt, "A nonlinear population Monte Carlo scheme for the Bayesian estimation of parameters of α-stable distributions," *Computational Statistics & Data Analysis*, vol. 95, pp. 57–74, March 2016.

[39] D. Williams, *Probability with martingales*, Cambridge University Press, Cambridge, (UK), 1991.

[40] M.-S. Oh and J. O. Breger, "Adaptive importance sampling in Monte Carlo integration," *Journal of Statistical Computation and Simulation*, vol. 41, no. 3-4, pp. 143–168, 1992.

[41] T. Bengtsson, P. Bickel, and B. Li, "Curse of dimensionality revisited: Collapse of particle filter in very large scale systems," *Probability and statistics: Essay in honour of David A. Freedman*, vol. 2, pp. 316–334, 2008.

[42] A. Beskos, D. Crisan, A. Jasra, et al., "On the stability of sequential Monte Carlo methods in high dimensions," *The Annals of Applied Probability*, vol. 24, no. 4, pp. 1396–1445, 2014.

[43] W.R. Gilks, S. Richardson, and D. Spiegelhalter, *Markov Chain Monte Carlo in Practice: Interdisciplinary Statistics*, Taylor & Francis, Inc., UK, 1995.

[44] S. Asmussen and P. W. Glynn, "A new proof of convergence of MCMC via the ergodic theorem," *Statistics & Probability Letters*, vol. 81, no. 10, pp. 1482–1485, 2011.

[45] C. P. Robert, *The Bayesian Choice*, Springer, 2007.

[46] L. Martino, V. Elvira, D. Luengo, and J. Corander, "MCMC-driven adaptive multiple importance sampling," in *Interdisciplinary Bayesian Statistics*, pp. 97–109. Springer, 2015.

[47] A. Lee, C. Yau, M. B. Giles, and C. C. Doucet, A.and Holmes, "On the utility of graphics cards to perform massively parallel simulation of advanced Monte Carlo methods," *Journal of Computational and Graphical Statistics*, vol. 19, no. 4, pp. 769–789, 2010.

[48] O. Hlinka, O. Slučiak, F. Hlawatsch, P. Djurić, and M. Rupp, "Likelihood consensus and its application to distributed particle filtering," *IEEE Transactions on Signal Processing*, vol. 60, no. 8, pp. 4334–4349, 2012.

[49] J. Quiñonero-Candela and C. E. Rasmussen, "A unifying view of sparse approximate Gaussian process regression," *Journal of Machine Learning Research*, vol. 6, no. Dec, pp. 1939–1959, 2005.

[50] R. J. Boys, D. J. Wilkinson, and T. B. L. Kirkwood, "Bayesian inference for a discretely observed stochastic kinetic model," *Statistics and Computing*, vol. 18, no. 2, pp. 125–135, 2008.

[51] C. Andrieu, A. Doucet, and R. Holenstein, "Particle Markov chain Monte Carlo methods," *Journal of the Royal Statistical Society B*, vol. 72, no. 3, pp. 269–342, 2010.

[52] C. K. Wikle, R. F. Milliff, D. Nychka, and L. M. Berliner, "Spatiotemporal hierarchical Bayesian modeling tropical ocean surface winds," *Journal of the American Statistical Association*, vol. 96, no. 454, pp. 382–397, 2001.

[53] J. Rougier, "Probabilistic inference for future climate using an ensemble of climate model evaluations," *Climatic Change*, vol. 81, no. 3, pp. 247–264, 2007.

[54] A. Lewis, "Efficient sampling of fast and slow cosmological parameters," *Physical Review D*, vol. 87, no. 10, pp. 103529, 2013.

[55] A. T. Ihler, J. W. Fisher, R. L. Moses, and A. S. Willsky, "Nonparametric belief propagation for self-localization of sensor networks," *IEEE Transactions on Selected Areas in Communications*, vol. 23, no. 4, pp. 809–819, April 2005.

[56] C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*, MIT Press, 2006.

[57] Y. Li and M. Coates, "Particle filtering with invertible particle flow," *arXiv preprint arXiv:1607.08799*, 2016.

[58] T. Homem-de Mello and G. Bayraksan, "Monte Carlo sampling-based methods for stochastic optimization," *Surveys in Operations Research and Management Science*, vol. 19, no. 1, pp. 56–85, 2014.

[59] T. Salimans, D. Kingma, and M. Welling, "Markov chain Monte Carlo and variational inference: Bridging the gap," in *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, 2015, pp. 1218–1226.