LAYERED ADAPTIVE IMPORTANCE SAMPLING

L. Martino[⋆]*, V. Elvira*[†]*, D. Luengo*[‡]*, J. Corander*[⋆]

[⋆] Dep. of Mathematics and Statistics, University of Helsinki, Helsinki (Finland).
[†] Dep. of Signal Theory and Communic., Universidad Carlos III de Madrid, Leganés (Spain).
[‡] Dep. of Circuits and Systems Engineering, Universidad Politécnica de Madrid, Madrid (Spain).

ABSTRACT

Monte Carlo algorithms represent the *de facto* standard for approximating complicated integrals involving multidimensional target distributions. In order to generate random realizations from the target distribution, Monte Carlo techniques use simpler proposal probability densities for drawing candidate samples. Performance of any such method is strictly related to the specification of the proposal distribution, such that unfortunate choices easily wreak havoc on the resulting estimators. In this work, we introduce a *layered*, that is a hierarchical, procedure for generating samples employed within a Monte Carlo scheme. This approach ensures that an appropriate equivalent proposal distribution is always obtained automatically (thus eliminating the risk of a catastrophic performance), although at the expense of a moderate increase in the complexity of the resulting algorithm. A hierarchical interpretation of two well-known methods, such as of the random walk Metropolis-Hastings (MH) and the Population Monte Carlo (PMC) techniques, is provided. Furthermore, we provide a general unified importance sampling (IS) framework where multiple proposal densities are employed, and several IS schemes are introduced applying the so-called deterministic mixture approach. Finally, given these schemes, we also propose a novel class of adaptive importance samplers using a population of proposals, where the adaptation is driven by independent parallel or interacting Markov Chain Monte Carlo (MCMC) chains. The resulting algorithms combine efficiently the benefits of both IS and MCMC methods.

Keywords: Bayesian Inference; Adaptive Importance Sampling; Population Monte Carlo; parallel MCMC.

1. INTRODUCTION

Monte Carlo methods currently represent a maturing toolkit widely used throughout science and technology [16, 39, 36]. Importance sampling (IS) and Markov Chain Monte Carlo (MCMC) methods are well-known Monte Carlo (MC) techniques applied to computing integrals involving a high-dimensional target probability density function (pdf) $\pi(\mathbf{x})$. In both cases, the choice of a suitable proposal density $q(\mathbf{x})$ is crucial for the success of the Monte Carlo based approximation. For this reason, the design of adaptive IS or MCMC schemes represents one of the most active research topics in this area and several methods have been proposed in literature [8, 12, 13, 20, 26].

Since both IS and MCMC have certain intrinsic advantages and weaknesses, several attempts have been made to successfully marry the two approaches: IS-within-MCMC [24, 25, 34] or MCMC-within-IS [3, 5, 11, 31, 33, 42]. To set the scene for such developments it is useful to recall briefly some main strengths of IS and MCMC, respectively. For instance, one benefit of IS is that it delivers a straightforward estimate [36, 23] of the normalizing constant of $\pi(\mathbf{x})$, which is also called evidence or marginal likelihood and is essential for several applications [19, 37]). However, estimation of the normalizing constant is highly challenging using MCMC and several authors have investigated different approaches to overcoming the obstacles related to instability of the resulting estimators [4, 6, 10, 19, 41]. Furthermore, the application and the theoretical analysis of an IS scheme using an adaptive proposal pdf is easier than of a corresponding adaptive MCMC scheme, where the theoretical analysis is more delicate.

On the other hand, an appealing feature of MCMC algorithms is their explorative behavior. For instance, the proposal function $q(\mathbf{x}|\mathbf{x}_{t-1})$ can depend on the previous state of the chain $\mathbf{x}_{t-1}$ and foster movement between different regions of the target density. For this reason, MCMC methods are usually preferred when no detailed information about the target $\pi(\mathbf{x})$ is available, especially in large dimensional spaces [22, 18, 2, 26]. A common feeling is that this intrinsic explorative nature seems to safeguard in some way the resulting Monte Carlo estimators with respect to a rough tuning of the proposal $q(\mathbf{x})$, explaining the evident wide success of the MCMC methods. In this work, we provide a framework for explaining this common feeling about the MCMC based on *random walk* proposal densities. In order to amplify their explorative behavior, several parallel MCMC chains can be run jointly [36, 23]. This strategy clearly fosters the exploration of the state space, at the expense of an

increasing computational cost. Several schemes have been introduced in order to share information among the different chains [9, 13, 28, 29], which further improves the overall convergence.

The first contribution of this work is a description and analysis of a hierarchical proposal procedure for generating samples, which are then employed within a Monte Carlo algorithm. In this hierarchical scheme, we consider two different conditionally independent levels: the first one is only used for generating location parameters for the proposal pdfs used in the second level for drawing possible candidates. Such an approach can be illustrated with an analogy of having a bag of potato chips/crisps layered on top of each other, such that their shapes and orientations may vary, mimicking a set of overlaid densities. We show that the *random walk Metropolis* [36] and the standard *Population Monte Carlo* (PMC) methods [8] can be interpreted as techniques which apply implicitly this hierarchical procedure. Furthermore, a second contribution of this work consists in providing a general framework for multiple importance sampling (MIS) schemes and their iterative adaptive versions. We discuss several alternative application of the so-called deterministic approach [38, 35] for sampling a mixture of pdfs. This general framework includes different MIS schemes used within AIS techniques already proposed in literature such as the standard PMC method [8], the adaptive multiple importance sampling [12, 27] and the adaptive population importance sampling [30], for instance.

Finally, we combine the general MIS framework with the hierarchical procedure for generating samples, introducing a new class of adaptive IS (AIS) algorithms. More specifically, one or several MCMC chains are used for driving an underlying MIS scheme. Each algorithm differs from the others in the specific Markov adaptation employed and the particular MIS technique applied for yielding the final Monte Carlo estimators. This novel class of algorithms combines efficiently the main strengths of the IS and the MCMC methods since it maintains an explorative behavior (as in MCMC) and can still easily estimate the normalizing constant (as in IS).

We describe in detail the simplest possible algorithm of this class, called *random walk importance sampling*. Moreover, we introduce two other different population-based variants for a specific choice of MIS scheme, which provides a good trade-off between performance and computational cost. In the first variant, the location parameters are updated according to several parallel MCMC chains. In the other one, an interacting adaptive strategy is applied. In both cases, all the adapted proposal pdfs collaborate to yield a single global IS estimator. One of the proposed algorithms, called *parallel interacting Markov adaptive importance sampling* (PI-MAIS), can be interpreted as parallel MCMC chains cooperating to produce a single global estimator, since the chains exchange statistical information to achieve common purpose.

The rest of the paper is organized as follows. Section 2 is devoted to recalling the problem statement. The hierarchical proposal procedure is introduced in Section 3. In Section 4, we describe a general framework for importance sampling schemes using a population of proposal pdfs. Section 5 introduces the adaptation procedure for updating the location parameters of these proposal pdfs. Numerical examples are provided in Section 6, including comparisons with several benchmark techniques. Different scenarios have been considered: multimodality and nonlinearity of the target distribution, as well as a positioning and tuning of parameters problem in a wireless sensor network. Finally, Section 7 contains some brief final remarks.

2. TARGET DISTRIBUTION AND RELATED INTEGRALS

In this work we focus on the Bayesian applications of IS and MCMC. However, the presented algorithms may be used for approximating any target distribution that needs to be handled by simulation methods. Let us denote a variable of interest as $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^{D_x}$, and let $\mathbf{y} \in \mathbb{R}^{D_y}$ be the observed data. The posterior pdf is then

$$\bar{\pi}(\mathbf{x}) = p(\mathbf{x}|\mathbf{y}) = \frac{\ell(\mathbf{y}|\mathbf{x})g(\mathbf{x})}{Z(\mathbf{y})}, \tag{1}$$

where $\ell(\mathbf{y}|\mathbf{x})$ is the likelihood function, $g(\mathbf{x})$ is the prior pdf and $Z(\mathbf{y})$ is the model evidence or partition function. In general, $Z(\mathbf{y})$ is unknown, so we consider the corresponding unnormalized target pdf,

$$\pi(\mathbf{x}) = \ell(\mathbf{y}|\mathbf{x})g(\mathbf{x}). \tag{2}$$

Our goal is computing efficiently an integral measure w.r.t. the target pdf,

$$I = \frac{1}{Z}\int_{\mathcal{X}} f(\mathbf{x})\pi(\mathbf{x})d\mathbf{x}, \tag{3}$$

where

$$Z = \int_{\mathcal{X}} \pi(\mathbf{x})d\mathbf{x}, \tag{4}$$

and f is a continuous function of $\mathbf{x}$. Since both $\bar{\pi}(\mathbf{x})$ and Z depend on the observations $\mathbf{y}$, the use of $\bar{\pi}(\mathbf{x}|\mathbf{y})$ and $Z(\mathbf{y})$ would be more precise. However, since the observations are fixed, in the sequel we remove the dependence on $\mathbf{y}$ to simplify the notation. In this work, we address the problem of approximating I and Z via Monte Carlo methods. Since drawing directly from $\pi(\mathbf{x}) \propto \bar{\pi}(\mathbf{x})$ is impossible in general, Monte Carlo techniques use a simpler proposal density for generating random candidates, testing or weighting them according to some proper suitable rule. We denote this normalized proposal pdf as $q(\mathbf{x})$. More specifically, we focus on the combined use of several proposal pdfs $q_1, \ldots, q_N$.

3. HIERARCHICAL PROCEDURE FOR PROPOSAL GENERATION

The performance of Monte Carlo methods depends strongly on the discrepancy between the target $\bar{\pi}(\mathbf{x}) \propto \pi(\mathbf{x})$ and the proposal $q(\mathbf{x})$. Namely, the performance improves if $q(\mathbf{x})$ is more similar (closer) to $\bar{\pi}(\mathbf{x})$. In general, tuning the parameters of the chosen proposal is a difficult task which requires statistical information of the target distribution. In this section, we deal with this important issue, focusing on the location parameter of the proposal pdf. Using a common formulation, we consider a proposal pdf defined by location $\boldsymbol{\mu}$ and scale $\mathbf{C}$ parameters, so that the proposal can be denoted as $q(\mathbf{x}|\boldsymbol{\mu}, \mathbf{C}) = q(\mathbf{x} - \boldsymbol{\mu}|\mathbf{C})$. We propose the following hierarchical procedure for generating samples employed afterwards within a Monte Carlo technique:

1. For $n = 1, \ldots, N$:
   (a) Draw a possible location parameter $\boldsymbol{\mu}_n \sim h(\boldsymbol{\mu})$.
   (b) Draw $\mathbf{x}_n^{(m)} \sim q(\mathbf{x}|\boldsymbol{\mu}_n, \mathbf{C})$, with $m = 1, \ldots, M$.
2. Use all the generated samples $\mathbf{x}_n^{(m)}$'s as candidates in a Monte Carlo method.

All the samples $\mathbf{x}_n^{(m)}$ are then used as candidates within a Monte Carlo technique. Note that $h(\boldsymbol{\mu})$ plays the role of a prior pdf over the location parameter of q. Each sample $\mathbf{x}_n^{(m)}$ has the following pdf

$$\widetilde{q}(\mathbf{x}|\mathbf{C}) = \int_{\mathcal{X}} q(\mathbf{x}|\boldsymbol{\mu}, \mathbf{C}) h(\boldsymbol{\mu}) d\boldsymbol{\mu}, \tag{5}$$

i.e., $\mathbf{x}_n^{(m)} \sim \widetilde{q}(\mathbf{x}|\mathbf{C})$ for all possible values of n and m. The density $\widetilde{q}$ is an *equivalent* proposal density corresponding to the hierarchical generating procedure. Note that the samples $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_N$, are not directly used in the Monte Carlo estimation, only $\mathbf{x}_n^{(m)}$ out of each pair $\{\boldsymbol{\mu}_n, \mathbf{x}_n^{(m)}\}$ enters the actual estimator. Hence, the immediate computational cost of the hierarchical approach is higher than in the standard approach, but as shown later, this can nevertheless imply substantial computational savings in terms of improved convergence towards the target. Furthermore, as shown above, we assume that the generation of $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_N$, is independent of the generated samples $\mathbf{x}_n^{(m)}$, $n = 1, \ldots, N$ and $m = 1, \ldots, M$.[1]

In contrast to the above hierarchical procedure, in standard adaptive Monte Carlo approches the parameter $\boldsymbol{\mu}_n$ is defined as a function $\delta : \mathbb{R}^{D_x \times (n-1)} \rightarrow \mathbb{R}^{D_x}$ of the previously generated samples $\mathbf{X}_{n-1} = [\mathbf{x}_1^{(1)}, \ldots, \mathbf{x}_1^{(M)}, \ldots, \mathbf{x}_{n-1}^{(1)}, \ldots, \mathbf{x}_{n-1}^{(M)}]$, i.e.,

$$\boldsymbol{\mu}_n = \delta(\mathbf{X}_{n-1}). \tag{6}$$

Moreover, the sequence $\boldsymbol{\mu}_1 \rightarrow \boldsymbol{\mu}_2 \rightarrow \ldots \rightarrow \boldsymbol{\mu}_N$ is typically converging to a fixed vector. In the hierarchical strategy, each $\boldsymbol{\mu}_n$ is always chosen randomly and independently of $\mathbf{X}_{n-1}$. Certain connections with other well-known Monte Carlo methods, such as the population Monte Carlo algorithm [8], are discussed in Sections 3.1.1 and 3.1.2. Note also that the complete generating procedure in Eq. (5) can be also interpreted as a data augmentation approach [23, 36], but we wish to emphasize the role of prior over the location parameters played by $h(\boldsymbol{\mu})$, for reasons that will become apparent later.

3.1. Optimizing prior for the location parameters

Assuming the parametric form of the proposal pdf $q(\mathbf{x}|\boldsymbol{\mu}, \mathbf{C})$ and its scale parameter $\mathbf{C}$ are chosen, we consider the problem of finding the optimal prior $h^*(\boldsymbol{\mu}|\mathbf{C})$ over the location parameter $\boldsymbol{\mu}$. The optimal prior depends on the chosen scale parameter $\mathbf{C}$ and since $q(\mathbf{x}|\boldsymbol{\mu}, \mathbf{C}) = q(\mathbf{x} - \boldsymbol{\mu}|\mathbf{C})$, as $\boldsymbol{\mu}$ is a location parameter, we can write

$$\widetilde{q}(\mathbf{x}|\mathbf{C}) = \int_{\mathcal{X}} q(\mathbf{x} - \boldsymbol{\mu}|\mathbf{C}) h^*(\boldsymbol{\mu}|\mathbf{C}) d\boldsymbol{\mu}. \tag{7}$$

[1] Note that, in the ideal case described here, each $\boldsymbol{\mu}_n$ is also independent of the other $\boldsymbol{\mu}$'s. However, in the rest of this work, we also consider the case with correlation among the location parameters $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_N$.

The desirable scenario is to have the equivalent proposal $\widetilde{q}(\mathbf{x}|\mathbf{C})$ coinciding exactly with the target $\bar{\pi}(\mathbf{x})$, i.e., $\widetilde{q}(\mathbf{x}|\mathbf{C}) = \bar{\pi}(\mathbf{x})$.[2] Eq. (7) can be rewritten in terms of the characteristic functions: $Q(\boldsymbol{\nu}|\mathbf{C}) = E[q(\mathbf{x}|\mathbf{C})e^{i\boldsymbol{\nu}\mathbf{x}}]$, $H^*(\boldsymbol{\nu}|\mathbf{C}) = E[h^*(\mathbf{x}|\mathbf{C})e^{i\boldsymbol{\nu}\mathbf{x}}]$ and $\bar{\Pi}(\boldsymbol{\nu}) = E[\bar{\pi}(\mathbf{x})e^{i\boldsymbol{\nu}\mathbf{x}}]$. Hence, the optimal prior pdf has the following characteristic function

$$H^*(\boldsymbol{\nu}|\mathbf{C}) = \frac{\bar{\Pi}(\boldsymbol{\nu})}{Q(\boldsymbol{\nu}|\mathbf{C})}. \tag{8}$$

In general, it is not possible to determine analytically the optimal prior pdf $h^*(\boldsymbol{\mu}|\mathbf{C})$, and thus, an efficient approximation is called for. For simplicity, here we set $M = 1$. Thus, we consider the generation of N samples $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, drawn following the previous hierarchical procedure, i.e., (a) draw a possible location parameter $\boldsymbol{\mu}_n \sim h(\boldsymbol{\mu})$ and (b) draw $\mathbf{x}_n \sim q(\mathbf{x}|\boldsymbol{\mu}_n, \mathbf{C})$. Observe that, in this procedure, we are using N different proposal pdfs

$$q(\mathbf{x}|\boldsymbol{\mu}_1, \mathbf{C}), \ldots, q(\mathbf{x}|\boldsymbol{\mu}_N, \mathbf{C}),$$

for drawing $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, where each $\mathbf{x}_n$ is drawn from n-th proposal $\mathbf{x}_n \sim q(\mathbf{x}|\boldsymbol{\mu}_n, \mathbf{C})$. Thus, we can interpreted that the set $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ is distributed according to the following mixture

$$\Phi(\mathbf{x}) = \frac{1}{N}\sum_{n=1}^{N} q(\mathbf{x}|\boldsymbol{\mu}_n, \mathbf{C}), \tag{9}$$

following the deterministic mixture argument [12]. Further details are given in Appendix B. The performance of the corresponding Monte Carlo method, where such a hierarchical procedure is applied, depends on how closely $\Phi(\mathbf{x})$ resembles $\bar{\pi}(\mathbf{x})$. If we choose $h(\boldsymbol{\mu}) = \bar{\pi}(\boldsymbol{\mu})$, i.e., the prior h is exactly the target then, since $\boldsymbol{\mu}_n \sim \bar{\pi}(\boldsymbol{\mu})$, and $\Phi(\mathbf{x})$ in Eq. (9) can be interpreted as a *kernel estimation* of $\bar{\pi}(\mathbf{x})$, where $q(\mathbf{x}|\boldsymbol{\mu}_1, \mathbf{C}), \ldots, q(\mathbf{x}|\boldsymbol{\mu}_N, \mathbf{C})$ play the role of kernel functions.

Therefore, $h(\boldsymbol{\mu}) = \bar{\pi}(\boldsymbol{\mu})$ is a good choice from a kernel density estimation point of view, but it is clearly infeasible in practice since we are not able to draw from $\bar{\pi}(\boldsymbol{\mu})$. One of our main ideas is therefore to apply another sampling method, such as an MCMC algorithm, to obtain the necessary samples $\{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_N\} \sim \bar{\pi}(\boldsymbol{\mu})$, for the other layer of the Monte Carlo. For instance, in Section 5 we design a general adaptive importance sampling (AIS) framework based on this idea, by combining MCMC with multiple importance sampling (MIS). With the choice $h(\boldsymbol{\mu}) = \bar{\pi}(\boldsymbol{\mu})$, the two levels of the sampler play different roles:

- The first level attends the need of *exploration* of the state space, providing $\{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_N\}$.
- The second level is devoted to the *approximation* of local features of the the target, using $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$.

In general, the two levels require their own tuning of the parameters of the corresponding proposal mechanisms. Two well-known Monte Carlo schemes, the random-walk Metropolis-Hastings (MH) [36] and the standard Population Monte Carlo (PMC) [8] methods, can be interpreted as *implicitly* using the hierarchical generating procedure with the prior $h(\boldsymbol{\mu}) = \bar{\pi}(\boldsymbol{\mu})$, which as exemplified in the next two subsections. However, there are some notable differences, as in both cases the generation of $\boldsymbol{\mu}$ is not independent of the previously generated $\mathbf{x}$. In random-walk MH, the two different sampling layers are "collapsed" into one, since in that case we have $\boldsymbol{\mu}_n = \mathbf{x}_n$. In the standard PMC technique it is possible to distinguish the two different layers, although the prior used is instead $h(\boldsymbol{\mu}) = \hat{\pi}^{(N)}(\boldsymbol{\mu})$, where $\hat{\pi}^{(N)}$ is an approximation of the measure of $\bar{\pi}(\boldsymbol{\mu})$ obtained using the previously generated samples $\mathbf{x}$ (in the second level of the hierarchical approach). The quality of this approximation increases with N, as discussed below and in Appendix D.

3.1.1. Hierarchical interpretation of the random walk Metropolis-Hastings algorithm

Consider the target $\pi(\mathbf{x}) \propto \bar{\pi}(\mathbf{x})$ and the proposal $q(\mathbf{x}|\mathbf{x}_{t-1}, \mathbf{C})$ where $\mathbf{x}_{t-1}$ the current state of the chain and $\mathbf{C}$ is a covariance matrix. One transition of the MH algorithm is summarized by

1. Draw $\mathbf{x}'$ from a proposal pdf $q(\mathbf{x}|\mathbf{x}_{t-1}, \mathbf{C})$.
2. Set $\mathbf{x}_t = \mathbf{x}'$ with probability
$$\alpha = \min\left[1, \frac{\pi(\mathbf{x}')q(\mathbf{x}_{t-1}|\mathbf{x}', \mathbf{C})}{\pi(\mathbf{x}_{t-1})q(\mathbf{x}'|\mathbf{x}_{t-1}, \mathbf{C})}\right],$$
otherwise set $\mathbf{x}_t = \mathbf{x}_{t-1}$ (with probability $1 - \alpha$).

[2] For instance, in an MCMC scheme, if $\widetilde{q}(\mathbf{x}|\mathbf{C}) = \bar{\pi}(\mathbf{x})$ then the Markov chain is formed by independent samples directly generated from $\bar{\pi}$.

There are two well-known general classes of proposal pdf: *independent proposal* $q(\mathbf{x}|\mathbf{x}_{t-1}, \mathbf{C}) = q(\mathbf{x}|\mathbf{C})$ (independent from the current state), and *random walk proposal* $q(\mathbf{x}|\mathbf{x}_{t-1}, \mathbf{C}) = q(\mathbf{x} - \mathbf{x}_{t-1}|\mathbf{C})$. Observe that in a random walk proposal $q(\mathbf{x}|\mathbf{x}_{t-1}, \mathbf{C})$, the current state $\mathbf{x}_{t-1}$ plays the role of the location parameter of q. The independent proposal strategy provides better performance than the random walk in terms of a smaller correlation among the generated samples, if certain prior information is available about the target so that all the parameters of q can be well-tuned. However, if the parameters of the independent proposal are not properly chosen, the performance of the algorithm easily deteriorates. In many cases no prior information about the target, such as location of the modes, mean or variance is available. For this reason, the use of a random walk proposal $q(\mathbf{x}|\mathbf{x}_{t-1}, \mathbf{C})$ is often preferred due to its explorative behavior, since it relocates the proposal at the current state of the chain at each iteration. As a consequence, the common wisdom is that this approach is more robust with respect to the choice of the tuning parameters. Below, we provide some further arguments explaining the success of the random walk approach.

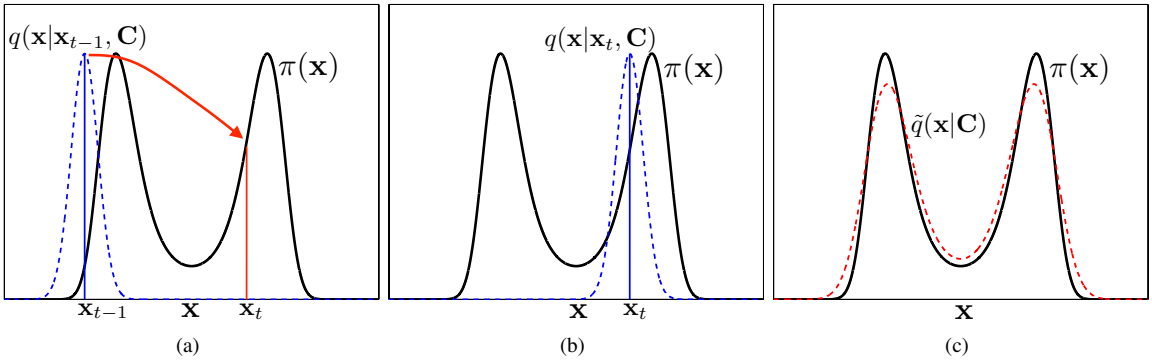


Fig. 1. Graphical representation of a random walk proposal and its equivalent independent proposal pdf. A bimodal target pdf $\pi(\mathbf{x})$ is shown in solid line. The proposal densities are depicted in dashed lines. **(a)** A proposal pdf $q(\mathbf{x}|\mathbf{x}_{t-1}, \mathbf{C})$ at the iteration $t-1$, and the next state of the chain $\mathbf{x}_t$. **(b)** The proposal pdf $q(\mathbf{x}|\mathbf{x}_t, \mathbf{C})$ at the t-th iteration. **(c)** The equivalent independent proposal pdf $\widetilde{q}(\mathbf{x}|\mathbf{C})$.

Consider the use of a random walk proposal density $q(\mathbf{x}|\mathbf{x}_{t-1}, \mathbf{C})$ in an MH algorithm. Furthermore, let us assume a "burn-in" length $T_b - 1$. Hence, considering an iteration $t \geq T_b$, we have $\mathbf{x}_t \sim \bar{\pi}(\mathbf{x})$, i.e., the chain has already reached the stationary (target) distribution. Therefore, for $t \geq T_b$, the probability of proposing a new sample using the random walk proposal $q(\mathbf{x} - \mathbf{x}_{t-1}|\mathbf{C})$ can be written as

$$\begin{aligned} \widetilde{q}(\mathbf{x}|\mathbf{C}) &= \int_{\mathcal{X}} q(\mathbf{x}|\mathbf{x}_{t-1}, \mathbf{C})\bar{\pi}(\mathbf{x}_{t-1})d\mathbf{x}_{t-1}, \\ &= \int_{\mathcal{X}} q(\mathbf{x} - \mathbf{x}_{t-1}|\mathbf{C})\bar{\pi}(\mathbf{x}_{t-1})d\mathbf{x}_{t-1}, \qquad \text{for } t \geq T_b, \end{aligned} \tag{10}$$

since $\mathbf{x}_{t-1} \sim \bar{\pi}(\mathbf{x}_{t-1})$ after the burn-in period, $t \geq T_b$, and $\mathbf{x}_{t-1}$ represents the location parameter of q. The function $\widetilde{q}(\mathbf{x}|\mathbf{C})$ is an *equivalent independent proposal* pdf corresponding to a random walk generating process within an MCMC method (after the "burn-in" period). It implies that the random walk generating process is equivalent, for $t \geq T_b$, to the following hierarchical procedure: (a) draw a location parameter $\boldsymbol{\mu}'$ from $\bar{\pi}(\boldsymbol{\mu})$, (b) draw $\mathbf{x}'$ from $q(\mathbf{x}|\boldsymbol{\mu}', \mathbf{C})$. Clearly, this alternative interpretation has no direct implications for practical purposes, since we are not able to draw directly form the target $\bar{\pi}$. However, it is useful for clarifying the main advantage of the random walk approach, i.e., that the equivalent proposal $\widetilde{q}$ is a better choice than an independent proposal roughly tuned by the user with non-optimal parameters. Indeed, the random walk generating procedure includes indirectly certain information about the target. Let be here $\mathbf{C}$ is a covariance matrix. Denoting $\mathbf{Z} \sim \widetilde{q}(\mathbf{x}|\mathbf{C})$, $\mathbf{S} \sim q(\mathbf{x}|\boldsymbol{\mu}, \mathbf{C})$ (assuming $E[\mathbf{S}] = \boldsymbol{\mu} = 0$), $\mathbf{X} \sim \bar{\pi}(\mathbf{x})$, we can write

$$E[\mathbf{Z}] = E[\mathbf{X}], \qquad \boldsymbol{\Sigma}_Z = \mathbf{C} + \boldsymbol{\Sigma}_X,$$

which are the mean and covariance matrix of $\mathbf{Z}$ with pdf $\widetilde{q}$. Moreover, after a finite number of iterations $t > T_b$, we can define an equivalent independent density $\widetilde{q}_T$ at the iteration T as

$$\widetilde{q}_T(\mathbf{x}|\mathbf{C}) = \frac{1}{T - T_b} \sum_{t=T_b}^{T} q(\mathbf{x} - \mathbf{x}_t|\mathbf{C}). \tag{11}$$

Since $\mathbf{x}_t \sim \bar{\pi}(\mathbf{x})$, for $t \geq T_b$, $\widetilde{q}_T(\mathbf{x}|\mathbf{C})$ is a kernel estimation of $\bar{\pi}(\mathbf{x})$ with kernel functions q. Clearly, $\widetilde{q}_T \to \widetilde{q}$ for $t \to +\infty$. Hence, the random walk generation process is equivalent to an independent proposal built via kernel estimation of the target. Figure 1 shows a graphical representation of equivalent proposal density $\widetilde{q}(\mathbf{x}|\mathbf{C})$. Observe that $\widetilde{q}(\mathbf{x}|\mathbf{C})$ is a much better proposal pdf than $q(\mathbf{x}|\boldsymbol{\mu}, \mathbf{C})$ with any possible choice of $\boldsymbol{\mu}$.

3.1.2. Hierarchical interpretation of Population Monte Carlo

A standard PMC method [8] is an adaptive importance sampler using a population of proposals $q_1, \ldots, q_N$. PMC consists of the following steps, given an initial set $\{\boldsymbol{\mu}_{1,0}, \ldots, \boldsymbol{\mu}_{N,0}\}$ of location parameters:

1. For $t = 1, \ldots, T$:

   (a) Draw $\mathbf{x}_{n,t} \sim q_n(\mathbf{x}|\boldsymbol{\mu}_{n,t-1}, \mathbf{C}_n)$, for $n = 1, \ldots, N$.

   (b) Assign to each sample $\mathbf{x}_{n,t}$ the weights,

   $$w_{n,t} = \frac{\pi(\mathbf{x}_{n,t})}{q_n(\mathbf{x}_{n,t}|\boldsymbol{\mu}_{n,t-1}, \mathbf{C}_n)}. \tag{12}$$

   (c) *Resampling:* draw N independent samples $\boldsymbol{\mu}_{1,t}, \ldots, \boldsymbol{\mu}_{N,t}$, according to the particle approximation

   $$\hat{\pi}_t^{(N)}(\boldsymbol{\mu}) = \frac{1}{\sum_{n=1}^N w_{n,t}} \sum_{n=1}^N w_{n,t} \delta(\boldsymbol{\mu} - \mathbf{x}_{n,t}). \tag{13}$$

   Note that each $\boldsymbol{\mu}_{n,t} \in \{\mathbf{x}_{1,t}, \ldots, \mathbf{x}_{N,t}\}$, with $n = 1, \ldots, N$.

2. Return all the pairs $\{\mathbf{x}_{n,t}, \bar{\rho}_{n,t}\}$ with $\bar{\rho}_{n,t} = \frac{w_{n,t}}{\sum_{t=1}^T \sum_{n=1}^N w_{n,t}}$, $n = 1, \ldots, N$ and $t = 1, \ldots, T$.

Fixing an iteration t, the generating procedure used in one iteration of the standard PMC method can be cast in the hierarchical formulation:

1. Draw N samples $\boldsymbol{\mu}_{1,t-1}, \ldots, \boldsymbol{\mu}_{N,t-1}$ from $\hat{\pi}_{t-1}^{(N)}(\boldsymbol{\mu})$, i.e., $\boldsymbol{\mu}_{n,t-1} \sim \hat{\pi}_{t-1}^{(N)}(\boldsymbol{\mu})$.

2. Draw $\mathbf{x}_{n,t} \sim q_n(\mathbf{x}|\boldsymbol{\mu}_{n,t-1}, \mathbf{C}_n)$, for $n = 1, \ldots, N$.

Note that $\hat{\pi}_t^{(N)}(\mathbf{x})$ is a particle approximation of $\bar{\pi}(\mathbf{x})$ that improves when N grows (see Appendix D for further details). Furthermore, the set of samples $\{\mathbf{x}_{1,t}, \ldots, \mathbf{x}_{N,t}\}$ can be interpreted as having been drawn in a more deterministic manner from the mixture $\frac{1}{N} \sum_{n=1}^N q_n(\mathbf{x}|\boldsymbol{\mu}_{n,t}, \mathbf{C}_n)$ (see Appendices B-C), such that

$$\{\mathbf{x}_{1,t}, \ldots, \mathbf{x}_{N,t}\} \sim \widetilde{q}(\mathbf{x}|\mathbf{C}_1, \ldots, \mathbf{C}_n) = \frac{1}{N} \sum_{n=1}^N q_n(\mathbf{x}|\boldsymbol{\mu}_{n,t-1}, \mathbf{C}_n), \tag{14}$$

although the standard version of PMC does not take advantages of this observation in the IS estimation. Since each $\boldsymbol{\mu}_{n,t-1} \sim \hat{\pi}_{t-1}^{(N)}(\boldsymbol{\mu})$, and $\hat{\pi}_{t-1}^{(N)}(\mathbf{x})$ is an approximation via IS of $\bar{\pi}(\mathbf{x})$, we can interpret $\widetilde{q}(\mathbf{x}|\mathbf{C}_1, \ldots, \mathbf{C}_n)$ as an approximate kernel density estimation of $\bar{\pi}$, where q_n's play the role of the kernel functions. The quality of this density estimation and, as a consequence, the performance of PMC depends on how well $\hat{\pi}_t^{(N)}$ approximates $\bar{\pi}$ (see Appendix D).

4. GENERALIZED MULTIPLE IMPORTANCE SAMPLING

In this section, we provide a general framework for multiple importance sampling (MIS) techniques using a population of proposal densities, which embeds various alternative samplers proposed in the literature. First, we consider several alternatives of static MIS, and then we focus on adaptive MIS schemes.

4.1. Generalized Static Multiple Importance Sampling

As we have already highlighted, finding a good proposal pdf $q(\mathbf{x})$ is critical and is in general very challenging [35]. An alternative and more robust strategy consists on using a population of different proposal pdfs. This scheme is often referred in the literature as *multiple importance sampling* (MIS) [8, 12, 30]. Consider a set of J (normalized) proposal pdfs,

$$q_1(\mathbf{x}), \ldots, q_J(\mathbf{x}),$$

with heavier tails than the target π, and let us assume that exactly M samples are drawn from each of them, i.e.,

$$\mathbf{x}_j^{(m)} \sim q_j(\mathbf{x}), \qquad j = 1, ..., J, \qquad m = 1, \ldots, M,$$

since we seek to having all the proposals participating in the IS estimation in order to increase the robustness of the Monte Carlo estimation. In this scenario, the importance weights associated to the samples can be obtained with one of the following strategies:

- *Standard* MIS (S-MIS):

$$w_j^{(m)} = \frac{\pi(\mathbf{x}_j^{(m)})}{q_j(\mathbf{x}_j^{(m)})}, \qquad j = 1, \ldots, J, \qquad m = 1, \ldots, M. \tag{15}$$

- *Deterministic mixture MIS* (DM-MIS) [35, 38]:

$$w_j^{(m)} = \frac{\pi(\mathbf{x}_j^{(m)})}{\psi(\mathbf{x}_j^{(m)})} = \frac{\pi(\mathbf{x}_j^{(m)})}{\frac{1}{J}\sum_{k=1}^{J} q_k(\mathbf{x}_j^{(m)})}, \quad j = 1, \ldots, J, \qquad m = 1, \ldots, M. \tag{16}$$

  where $\psi(\mathbf{x}) = \frac{1}{J}\sum_{j=1}^{J} q_j(\mathbf{x})$ is the mixture pdf, composed of all the proposal pdfs. This approach interprets the complete set of samples, $\{\mathbf{x}_j\}_{j=1}^{J}$, as being distributed according to the mixture $\psi(\mathbf{x})$, i.e., $\{\mathbf{x}_1, \ldots, \mathbf{x}_J\} \sim \psi(\mathbf{x})$. See Appendices B and C for further details.

In both cases, the consistency of the estimators is ensured. The main advantage of the DM-MIS weights is that they yield more stable and efficient estimators [17, 35]. However, the DM-MIS estimator is computationally more expensive, as it requires J evaluations of the proposal pdfs to obtain each weight instead of just one. In total, JM evaluations of proposals are required. Note that the number of evaluations of the target $\pi(\mathbf{x})$ is the same regardless of whether the weights are calculated according to (15) or (16). In some cases this additional computational load may be excessive (especially for large values of J) and alternative efficient solutions are desirable. For instance, following the argument in Appendix B, partial different mixtures can be considered [17]. As an example:

- *Partial* DM-MIS (P-DM-MIS) [17]: divide the J proposals in $L = \frac{J}{P}$ disjoint groups forming L mixtures with P components. We denote the set of P indices corresponding to the ℓ-th mixture ($\ell = 1, \ldots, L$) as $\mathcal{S}_\ell = \{k_{\ell,1}, \ldots, k_{\ell,P}\}$ (hence, $|\mathcal{S}_\ell| = P$) where each $k_{\ell,p} \in \{1, \ldots, J\}$. Thus, we have

$$\mathcal{S}_1 \cup \mathcal{S}_2 \cup \ldots \cup \mathcal{S}_L = \{1, \ldots, J\}, \tag{17}$$

  with $\mathcal{S}_r \cap \mathcal{S}_\ell = \emptyset$ for all $\ell = 1, \ldots, L$ and $r \neq \ell$. Therefore, in this case, the importance weights are defined as

$$w_j^{(m)} = \frac{\pi(\mathbf{x}_j^{(m)})}{\frac{1}{P}\sum_{k \in \mathcal{S}_\ell} q_k(\mathbf{x}_j^{(m)})}, \quad j \in \mathcal{S}_\ell, \quad \ell = 1, \ldots, L, \quad m = 1, \ldots, M. \tag{18}$$

In the next subsection, we describe a framework where a partial grouping of the proposal pdfs arises naturally from the sampler definition. All the previous cases can be captured by a generic mixture-proposal pdf $\Phi_j(\mathbf{x})$ under which the MIS weight can be defined as

$$w_j^{(m)} = \frac{\pi(\mathbf{x}_j^{(m)})}{\Phi_j(\mathbf{x}_j^{(m)})}, \qquad j = 1, \ldots, J, \qquad m = 1, \ldots, M, \tag{19}$$

Table 1. Summary of the possible functions $\Phi_j(\mathbf{x})$ for MIS strategies.

MIS approach	**Function** $\Phi_j(\mathbf{x})$, $(j = 1, \ldots, J)$	**L**	**P**
		$LP = J$	
Standard MIS	$q_j(\mathbf{x})$	J	1
DM-MIS	$\psi(\mathbf{x}) = \frac{1}{J}\sum_{j=1}^{J} q_j(\mathbf{x})$	1	J
Partial DM-MIS	generic mixture $\Phi_j(\mathbf{x})$	L	P

Table 2. Generic static MIS scheme.

1. **Generation:** Draw M samples from each q_j, i.e.,
$$\mathbf{x}_j^{(m)} \sim q_j(\mathbf{x}),$$
with $j = 1, \ldots, J$ and $m = 1, \ldots, M$.
2. **Weighting:** Assign to the sample $\mathbf{x}_j^{(m)}$ the following weight
$$w_j^{(m)} = \frac{\pi(\mathbf{x}_j^{(m)})}{\Phi_j(\mathbf{x}_j^{(m)})}, \qquad j = 1, \ldots, J, \quad m = 1, \ldots, M, \tag{21}$$
where Φ_j is a finite mixture of q_j's (with equal weights), as shown in Table 1.
3. **Normalization:** Set
$$\bar{\rho}_j^{(m)} = \frac{w_j^{(m)}}{\sum_{j=1}^{J}\sum_{m=1}^{M} w_j^{(m)}}.$$
4. **Output:** Return all the pairs $\{\mathbf{x}_j^{(m)}, \bar{\rho}_j^{(m)}\}$, for $j = 1, \ldots, J$ and $m = 1, \ldots, M$.

where $\Phi_j(\mathbf{x}_j^{(m)}) = q_j(\mathbf{x}_j^{(m)})$ in Eq. (15), $\Phi_j(\mathbf{x}_j^{(m)}) = \frac{1}{J}\sum_{k=1}^{J} q_k(\mathbf{x}_j^{(m)})$ in Eq. (16) and $\Phi_j(\mathbf{x}_j^{(m)}) = \frac{1}{P}\sum_{k\in\mathcal{S}_\ell} q_k(\mathbf{x}_j^{(m)})$ with $j \in \mathcal{S}_\ell$ in Eq. (18). The weights must normalized as

$$\bar{\rho}_j^{(m)} = \frac{w_j^{(m)}}{\sum_{j=1}^{J}\sum_{m=1}^{M} w_j^{(m)}}. \tag{20}$$

Table 1 shows the different possible choices of $\Phi_j(\mathbf{x}_j^{(m)})$, whereas Table 2 summarizes a generic static MIS procedure.

Note that the IS estimation $\hat{I}$ of a specific moment of $\bar{\pi}$, i.e., the integral I given in Eq. (3), and the approximation $\hat{Z}$ of the normalizing constant in Eq. (4), are now expressed as

$$\hat{I} = \sum_{j=1}^{J}\sum_{m=1}^{M} \bar{\rho}_j^{(m)} f(\mathbf{x}_j^{(m)}), \quad \hat{Z} = \frac{1}{JM}\sum_{j=1}^{J}\sum_{m=1}^{M} w_j^{(m)}. \tag{22}$$

Then, the particle approximation of the measure of $\bar{\pi}$ is given by

$$\hat{\pi}^{(J)}(\mathbf{x}) = \frac{1}{\hat{Z}}\sum_{j=1}^{J}\sum_{m=1}^{M} w_j^{(m)}\delta(\mathbf{x} - \mathbf{x}_j^{(m)}). \tag{23}$$

4.2. Generalized Adaptive Multiple Importance Sampling

In order to decrease the mismatch between the proposal and target pdfs, several Monte Carlo methods adapt iteratively the parameters of the proposal pdf using the information of the past samples [8, 12, 30]. In the adaptive scenario, we have a set of proposal pdfs $\{q_{n,t}(\mathbf{x})\}$, with $n = 1, \ldots, N$ and $t = 1, \ldots, T$, where the subscript t indicates the iteration index and T is the total number of adaptation steps. Here $J = NT$ is the total number of proposal pdfs. Here we present a general unified framework, called generalized adaptive multiple importance sampling (GAMIS), which includes several methodologies

proposed separately in literature. In GAMIS, each proposal pdf in the population $\{q_{n,t}\}$ is updated at each iteration $t = 1, \ldots, T$, forming the sequence

$$q_{n,1}(\mathbf{x}), q_{n,2}(\mathbf{x}), \ldots, q_{n,T}(\mathbf{x}),$$

for the n-th proposal. A graphical characterization of this process is shown in Figure 2. At the t-th iteration, the adaptation procedure takes into account certain statistical information about the target distribution achieved in the previous iteration $t = 1, \ldots, t-1$ (for this purpose several procedures have been proposed, for instance see [7, 8, 12, 30]). Furthermore, at the t-th iteration, M samples are drawn from each proposal $q_{n,t}$, i.e.,

$$\mathbf{x}_{n,t}^{(m)} \sim q_{n,t}(\mathbf{x}), \quad \text{with} \quad m = 1, \ldots, M,$$

$n = 1, \ldots, N$ and $t = 1, \ldots, T$. To each sample $\mathbf{x}_{n,t}^{(m)}$, an importance weight $w_{n,t}^{(m)}$ is assigned. Several strategies can be applied to build $w_{n,t}^{(m)}$ considering the different MIS approaches. Figure 2 provides a graphical representation of this scenario, by showing both the spatial and temporal evolution of the $J = NT$ proposal pdfs.

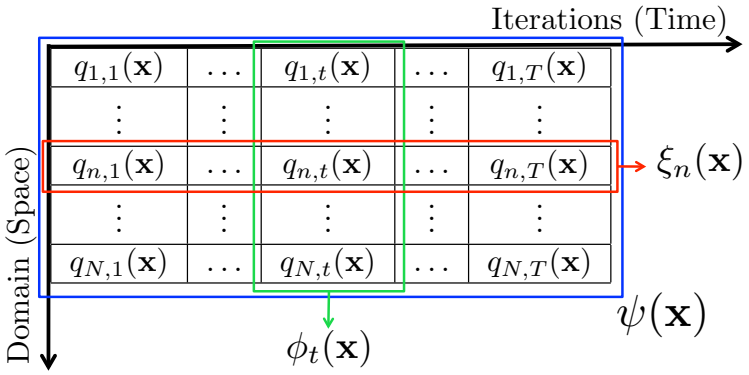


Fig. 2. Graphical representation of the $J = NT$ proposal pdfs used in a generic adaptive multiple IS scheme, spread in the domain $\mathcal{X}$ ($n = 1, \ldots, N$) and adapted over the time ($t = 1, \ldots, T$). There are 3 possible kind of mixtures displayed: $\psi(\mathbf{x})$ involving all the proposals, $\phi_t(\mathbf{x})$ involving only the proposals at the iteration t and $\xi_n(\mathbf{x})$ considering the temporal evolution of the n-th proposal pdf.

In any possible AIS algorithm, one weight

$$w_{n,t}^{(m)} = \frac{\pi(\mathbf{x}_{n,t}^{(m)})}{\Phi_{n,t}(\mathbf{x}_{n,t}^{(m)})}, \tag{24}$$

is associated to each sample $\mathbf{x}_{n,t}^{(m)}$. In a standard MIS approach, the function employed in the weight denominator is

$$\Phi_{n,t}(\mathbf{x}) = q_{n,t}(\mathbf{x}). \tag{25}$$

In the complete DM-MIS case, the function $\Phi_{n,t}$ is defined as

$$\Phi_{n,t}(\mathbf{x}) = \psi(\mathbf{x}) = \frac{1}{NT} \sum_{k=1}^{N} \sum_{r=1}^{T} q_{k,r}(\mathbf{x}). \tag{26}$$

This case corresponds to the blue rectangle in Fig. 2. Moreover, two clear possibilities of partial DM-MIS schemes appear in this scenario. The first one uses the following partial mixture

$$\Phi_{n,t}(\mathbf{x}) = \xi_n(\mathbf{x}) = \frac{1}{T} \sum_{r=1}^{T} q_{n,r}(\mathbf{x}), \quad \text{for} \quad n = 1, \ldots, N. \tag{27}$$

as mixture-proposal pdf in the IS weight denominator. Namely, in this case we consider the temporal evolution of the n-th single proposal $q_{n,t}$. Hence, we have $L = N$ mixtures, each one formed by $P = T$ components (see red rectangle in Fig. 2).

The other possibility consists in considering the mixture of all the $q_{n,t}$'s at the iteration t, i.e.,

$$\Phi_{n,t}(\mathbf{x}) = \phi_t(\mathbf{x}) = \frac{1}{N}\sum_{k=1}^{N} q_{k,t}(\mathbf{x}), \quad \text{for} \quad t = 1, \ldots, T, \tag{28}$$

so that we have $L = T$ mixtures, each one formed by $P = N$ components (see green rectangle in Fig. 2). The function $\Phi_{n,t}$ in Eq. (25) is used in the standard PMC scheme [8]; the case in Eq. (27) with $N = 1$ has been considered in the adaptive multiple importance sampling (AMIS) [12]. The choice in Eq. (28) has been applied in the adaptive population importance sampling (APIS) algorithm [30] and in [7, 14, 15] a similar strategy is employed but using a standard (non-deterministic) sampling of the mixture $\phi_t(\mathbf{x})$.

Table 3 summarizes the discussed possible cases. The last row corresponds to a different (generic) grouping strategy of the proposal pdfs $q_{n,t}$. As previously described, we can also divide the $J = NT$ proposals in $L = \frac{NT}{P}$ disjoint groups forming L mixtures with P components. We denote the set of P pairs of indices corresponding to the ℓ-th mixture ($\ell = 1, \ldots, L$) as $\mathcal{S}_\ell = \{(k_{\ell,1}, r_{\ell,1}), \ldots, (k_{\ell,P}, r_{\ell,P})\}$ where each $k_{\ell,p} \in \{1, \ldots, N\}$ and $r_{\ell,p} \in \{1, \ldots, T\}$ (hence, $|\mathcal{S}_\ell| = P$ where each element is a pair of indices) . In this scenario, we have

$$\Phi_{n,t}(\mathbf{x}) = \frac{1}{P}\sum_{(k,r)\in\mathcal{S}_\ell} q_{k,r}(\mathbf{x}), \quad \text{with } (n,t) \in \mathcal{S}_\ell, \quad \text{for} \quad \ell = 1, \ldots, L. \tag{29}$$

Note that using $\psi(\mathbf{x})$ and $\xi_n(\mathbf{x})$ the computational cost increases as the total number of iterations T grows. Indeed, at the generic t-th iteration, all the previous proposals $q_{n,1}, \ldots, q_{n,t-1}$ (for all n) must be evaluated at all the new samples $\mathbf{x}_{n,t}^{(m)}$. Using $\phi_t(\mathbf{x})$, the computational cost is controlled by N, regardless of the number of performed adaptive steps.

Table 3. Summary of possible MIS strategies in an adaptive framework.

MIS approach	**Function** $\Phi_{n,t}(\mathbf{x})$	**J**	**L**	**P**	**Corresponding Algorithm**
			$LP = J$		
Standard MIS	$q_{n,t}(\mathbf{x})$	NT	NT	1	Stand. AIS [36] and Stand. PMC [8]
DM-MIS	$\psi(\mathbf{x}) = \frac{1}{NT}\sum_{n=1}^{N}\sum_{t=1}^{T} q_{n,t}(\mathbf{x})$		1	NT	suggested in [17]
Partial DM-MIS	$\xi_n(\mathbf{x}) = \frac{1}{T}\sum_{t=1}^{T} q_{n,t}(\mathbf{x})$		N	T	AMIS [12], with $N = 1$
Partial DM-MIS	$\phi_t(\mathbf{x}) = \frac{1}{N}\sum_{n=1}^{N} q_{n,t}(\mathbf{x})$		T	N	APIS [30] and [7, 14, 15]
Partial DM-MIS	generic $\Phi_{n,t}(\mathbf{x})$ in Eq. (29)		L	P	suggested in [17]

Observe that a suitable AIS scheme builds iteratively a global IS estimator which uses the final normalized weights

$$\bar{\rho}_{n,t}^{(m)} = \frac{w_{n,t}^{(m)}}{\sum_{t=1}^{T}\sum_{n=1}^{N}\sum_{m=1}^{M} w_{n,t}^{(m)}}, \quad n = 1, \ldots, N, \quad m = 1, \ldots, M, \quad t = 1, \ldots, T. \tag{33}$$

It is important to remark that a GAMIS scheme can be applied in two different ways:

- *Batch mode:* all the adaptation steps can be performed in advance, generating the entire population of $J = NT$ proposal pdfs $\{q_{n,t}\}$'s. After that, the algorithm is converted in a simpler static MIS technique. The importance weights are computed and normalized as in Table 2. This version is simpler than the iterative mode described below, but the output of the algorithm is provided only after adapting all the proposals, i.e., after T iterations.
- *Iterative mode:* a GAMIS scheme can be formulated in a iterative way providing an output at each iteration t. Table 4 shows this iterative version. It is important to remark that, at the t-th iteration, the weights of the samples generated previously need to be recalculated, as shown at step 2(c-3) in Table 4. The choices of $\Phi_{n,t}(\mathbf{x}) = q_{n,t}(\mathbf{x})$ or $\Phi_{n,t}(\mathbf{x}) = \phi_t(\mathbf{x})$ allow one to avoid completely this re-computation step of the weights.

For simplicity, in Table 4, we have provided the output of the algorithms as weighted samples, i.e., all the pairs $\{\mathbf{x}_{n,t}^{(m)}, \bar{\rho}_{n,t}^{(m)}\}$. However, the output can be equivalently expressed as an estimation of a specific moment of the target. Indeed, in this case, the final global IS estimations $\hat{I}_T$ and $\hat{Z}_T$ are

$$\hat{I}_T = \sum_{\tau=1}^{T}\sum_{n=1}^{N}\sum_{m=1}^{M} \bar{\rho}_{n,\tau}^{(m)} f(\mathbf{x}_{n,\tau}^{(m)}), \qquad \hat{Z}_T = \frac{1}{NMT}\sum_{\tau=1}^{T}\sum_{n=1}^{N}\sum_{m=1}^{M} w_{n,\tau}^{(m)}, \tag{34}$$

Table 4. Generic GAMIS scheme: *iterative* version.

1. **Initialization:** Set $t = 1$, $H_0 = 0$ and choose initial N proposal pdfs $q_{n,0}(\mathbf{x})$.
2. For $t = 1, \ldots, T$:
   (a) **Adaptation:** update the proposal pdfs $\{q_{n,t-1}\}_{n=1}^N$ providing $\{q_{n,t}\}_{n=1}^N$, taking in account the information of the previous generated samples $\mathbf{x}_{n,\tau}^{(m)}$, with $\tau = 1, \ldots, t-1$, $n = 1, \ldots, N$ and $m = 1, \ldots, M$ (see [8, 7, 12, 30] for some specific adaptive algorithms).
   (b) **Generation:** Draw M samples from each $q_{n,t}$, i.e., $\mathbf{x}_{n,t}^{(m)} \sim q_{n,t}(\mathbf{x})$, with $n = 1, \ldots, N$ and $m = 1, \ldots, M$.
   (c) **Weighting:**
      (c-1) Update the function $\Phi_{n,t}(\mathbf{x})$ given the current population $\{q_{1,t}, \ldots, q_{N,t}\}$.
      (c-2) Assign the weights to the new samples $\mathbf{x}_{n,t}^{(m)}$,

$$w_{n,t}^{(m)} = \frac{\pi(\mathbf{x}_{n,t}^{(m)})}{\Phi_{n,t}(\mathbf{x}_{n,t}^{(m)})}, \qquad n = 1, \ldots, N, \text{ and } m = 1, \ldots, M. \tag{30}$$

      (c-3) Re-weight the previous samples $\mathbf{x}_{n,\tau}^{(m)}$ for $\tau = 1, \ldots, t-1$ as

$$w_{n,\tau}^{(m)} = \frac{\pi(\mathbf{x}_{n,\tau}^{(m)})}{\Phi_{n,t}(\mathbf{x}_{n,\tau}^{(m)})}, \qquad \tau = 1, \ldots, t-1, \quad n = 1, \ldots, N, \text{ and } m = 1, \ldots, M. \tag{31}$$

   (d) **Normalization:** Set $S_t = \sum_{m=1}^M \sum_{n=1}^N w_{n,t}^{(m)}$, $H_t = H_{t-1} + S_t$, and normalize all the weights (so far),

$$\bar{\rho}_{n,\tau}^{(m)} = \bar{\rho}_{n,\tau-1}^{(m)} \frac{H_{t-1}}{H_t}, \quad \tau = 1, \ldots, t, \quad n = 1, \ldots, N, \quad m = 1, \ldots, M. \tag{32}$$

   (e) **Output:** Return all the pairs $\{\mathbf{x}_{n,\tau}^{(m)}, \bar{\rho}_{n,\tau}^{(m)}\}$, for $\tau = 1, \ldots, t$, $n = 1, \ldots, N$, and $m = 1, \ldots, M$.

where $\bar{\rho}_{n,\tau}^{(m)} = \frac{w_{n,\tau}^{(m)}}{\hat{Z}_T}$. Moreover, the particle approximation of the measure of $\bar{\pi}$ is

$$\hat{\pi}^{(NMT)}(\mathbf{x}) = \frac{1}{\hat{Z}_T} \sum_{\tau=1}^{T} \sum_{n=1}^{N} \sum_{m=1}^{M} w_{n,\tau}^{(m)} \delta(\mathbf{x} - \mathbf{x}_{n,\tau}^{(m)}). \tag{35}$$

Eqs. (34) can be expressed recursively providing an estimation at each iteration t. Starting with $H_0 = 0$, $\hat{I}_0 = 0$, and setting $S_t = \sum_{n=1}^N \sum_{m=1}^M w_{n,t}^{(m)}$, $H_t = H_{t-1} + S_t$, we have

$$\begin{aligned} \hat{I}_t &= \frac{1}{H_t} \left[H_{t-1} \hat{I}_{t-1} + \sum_{n=1}^{N} \sum_{m=1}^{M} w_{n,t}^{(m)} f(\mathbf{x}_{n,t}^{(m)}) \right], \\ \hat{I}_t &= \frac{H_{t-1}}{H_{t-1} + S_t} \hat{I}_{t-1} + \frac{S_t}{H_{t-1} + S_t} \hat{A}_t, \end{aligned} \tag{36}$$

where $\hat{A}_t = \sum_{n=1}^N \sum_{m=1}^M \frac{w_{n,t}^{(m)}}{S_t} f(\mathbf{x}_{n,t}^{(m)})$ is a partial IS estimator using only the samples drawn at the t-th iterations. Therefore, $\hat{I}_t$ can be seen as a convex combination of the two IS estimators $\hat{I}_{t-1}$ and $\hat{A}_t$ (for further explanations see Eqs. (46)- (47) in App. C.1). Observe that, the final global estimator $\hat{I}_T$ in Eq. (34), obtained recursively in Eq. (36), is simply a standard IS estimator using all the samples $\mathbf{x}_{n,t}^{(m)}$ and considering the mixtures $\Phi_{n,t}(\mathbf{x})$ as proposal pdfs in the IS weight ratio. Finally, note that

$$\hat{Z}_t = \frac{1}{t} \frac{1}{NM} H_t. \tag{37}$$

A brief discussion about the consistency of $\hat{I}_t$ and $\hat{Z}_t$ is given in Appendix A.

5. MARKOV ADAPTATION FOR GAMIS

In this section, we combine the two general ideas discussed previously, in order to design efficient AIS techniques. More specifically, we apply the hierarchical Monte Carlo approach to adapt the proposal pdfs within a GAMIS scheme. Therefore, a

Markov GAMIS technique consists on the two following layers:

1. *Upper level (Adaptation):* Given the set of location parameters,

$$\mathcal{P}_{t-1} = \{\boldsymbol{\mu}_{1,t-1}, \ldots, \boldsymbol{\mu}_{N,t-1}\},$$

   provide the new set $\mathcal{P}_t = \{\boldsymbol{\mu}_{1,t}, \ldots, \boldsymbol{\mu}_{N,t}\}$ according to MCMC transitions with $\bar{\pi}$ as invariant density.

2. *Lower level (MIS estimation):* Given the population of proposals

$$q_{1,t}(\mathbf{x}|\boldsymbol{\mu}_{1,t}, \mathbf{C}_1),\ q_{2,t}(\mathbf{x}|\boldsymbol{\mu}_{2,t}, \mathbf{C}_2), \ldots, q_{N,t}(\mathbf{x}|\boldsymbol{\mu}_{N,t}, \mathbf{C}_N),$$

   choose a function $\Phi_{n,t}(\mathbf{x})$ employed in the computation of the weights in Eq. (24), and perform a MIS approximation of the target as described in Section 4.2.

The theoretical motivation of this adaptation procedure is supported by the previously discussed kernel estimation argument. Observe that the adaptation process is independent from the underlying IS steps. Markov GAMIS is a general framework which contains several possible algorithms, depending on the MCMC strategy used for updating the location parameters and the specific choice of the function $\Phi_{n,t}$. Table 5 provides several examples of possible novel techniques determined by the value of N, the choice of $\Phi_{n,t}$, and the type of MCMC adaptation. Some of them are variants of well-known techniques as PMC [8] and AMIS [12] where the Markov adaptation procedure is employed. Others, such as the *Random Walk Importance Sampling* (RWIS), the *Parallel Interacting Markov Adaptive Importance Sampling* (PI-MAIS) and *Doubly Interacting Markov Adaptive Importance Sampling* (I^2-MAIS) are described below in detail. For these novel algorithms, we have set $\Phi_{n,t}(\mathbf{x}) = \phi_t(\mathbf{x})$ so that the computational cost is directly controlled by the parameter N and, therefore, the re-weighting step 2(c-3) in Table 4 is not required.

RWIS is the simplest possible Markov GAMIS algorithm. Specifically, for the MCMC adaptation we consider a standard MH technique, setting $N = 1$ and choosing $\Phi_{n,t}(\mathbf{x}) = \phi_t(\mathbf{x}) = q_{n,t}(\mathbf{x})$ (since $N = 1$, the two cases coincide). Table 6 shows the RWIS algorithm. Note that we can distinguish the proposal pdf used in MH, $\varphi(\boldsymbol{\mu}|\boldsymbol{\mu}_{t-1}, \boldsymbol{\Lambda})$, from the proposal pdf used in IS part, $q(\mathbf{x}|\boldsymbol{\mu}_t, \mathbf{C})$. As described in the general motivation, there are now two different proposal densities, one at each level of the hierarchical Monte Carlo. The MH technique is applied to obtain good location parameters for the underlying IS and the particle approximation of $\bar{\pi}$ is then obtained iteratively using NT samples.

Table 5. Example of possible Markov GAMIS algorithms.

		Parallel adaptation	**Interacting adaptation**
Function $\boldsymbol{\Phi}_{n,t}(\mathbf{x})$	$\mathbf{N = 1}$	$\mathbf{N > 1}$	$\mathbf{N > 1}$
$q_{n,t}(\mathbf{x})$	RWIS (see Table 6)	Markov PMC (related to [8])	
$\xi_n(\mathbf{x}) = \frac{1}{T}\sum_{t=1}^{T} q_{n,t}(\mathbf{x})$	Markov AMIS (related to [12])	N parallel Markov AMIS (rel. to [12])	Population-based Markov AMIS (rel. to [12])
$\phi_t(\mathbf{x}) = \frac{1}{N}\sum_{n=1}^{N} q_{n,t}(\mathbf{x})$	RWIS (see Table 6)	PI-MAIS (see Section 5.1)	I^2-MAIS (see Section 5.1)
$\psi(\mathbf{x}) = \frac{1}{NT}\sum_{n=1}^{N}\sum_{t=1}^{T} q_{n,t}(\mathbf{x})$	Markov AMIS (related to [12])	Full Markov GAMIS	
generic $\Phi_{n,t}(\mathbf{x})$	Partial Markov GAMIS		

5.1. Population-based algorithms

The RWIS technique can be easily extended using a population of N proposal pdfs. We again choose $\Phi_{n,t}(\mathbf{x}) = \phi_t(\mathbf{x}) = \frac{1}{N}\sum_{n=1}^{N} q_{n,t}(\mathbf{x})$, so that the computational cost depends only on N, independently of the total number of iterations T. Moreover, step 2(c-3) in Table 4 is not required, in this case. Table 7 describes the corresponding algorithm without specifying the MCMC approach used for generating the cloud of the new parameters $\mathcal{P}_t = \{\boldsymbol{\mu}_{1,t}, ..., \boldsymbol{\mu}_{N,t}\}$ given the previous set $\mathcal{P}_{t-1}$. Clearly, RWIS is a special case of the algorithm in Table 4 when $N = 1$.

Table 6. Random Walk Importance Sampling (RWIS) algorithm.

1. **Initialization:** start with $t = 1$, $H_0 = 0$, choose the values M and T, the initial location parameter $\boldsymbol{\mu}_0$, and the scale parameter $\mathbf{C}$ and $\boldsymbol{\Lambda}$.
2. For $t = 1, \ldots, T$:
   (a) **MH step:**
   (a-1) Draw $\boldsymbol{\mu}'$ from a proposal pdf $\varphi(\boldsymbol{\mu}|\boldsymbol{\mu}_{t-1}, \boldsymbol{\Lambda})$.
   (a-2) Set $\boldsymbol{\mu}_t = \boldsymbol{\mu}'$ with probability
   $$\alpha = \min\left[1, \frac{\pi(\boldsymbol{\mu}')\varphi(\boldsymbol{\mu}_t|\boldsymbol{\mu}', \boldsymbol{\Lambda}),}{\pi(\boldsymbol{\mu}_t)\varphi(\boldsymbol{\mu}'|\boldsymbol{\mu}_{t-1}, \boldsymbol{\Lambda})}\right],$$
   otherwise set $\boldsymbol{\mu}_t = \boldsymbol{\mu}_{t-1}$ (with probability $1 - \alpha$).
   (b) **IS steps:**
   (b-1) Draw $\mathbf{x}_t^{(m)} \sim q_t(\mathbf{x}|\boldsymbol{\mu}_t, \mathbf{C}_n)$ with $m = 1, \ldots, M$.
   (b-2) Weight the samples with
   $$w_t^{(m)} = \frac{\pi(\mathbf{x}_t^{(m)})}{q_t(\mathbf{x}_t^{(m)}|\boldsymbol{\mu}_t, \mathbf{C}_n)},$$
   (b-3) Set $S_t = \sum_{m=1}^{M} w_t^{(m)}$, $H_t = H_{t-1} + S_t$, and normalize the weights
   $$\bar{\rho}_t^{(m)} = \frac{w_t^{(m)}}{\sum_{\tau=1}^{t}\sum_{m=1}^{M} w_t^{(m)}} = \bar{\rho}_{t-1}^{(m)} \frac{H_{t-1}}{H_t}.$$
   (c) **Output:** Return all the pairs $\{\mathbf{x}_\tau^{(m)}, \bar{\rho}_\tau^{(m)}\}$ for $m = 1, \ldots, M$ and $\tau = 1, \ldots, t$.

Two possible adaptation procedures via MCMC are discussed below. In the first one, we consider N independent parallel chains for updating the N location parameters. We refer to this method as Parallel Interacting Markov Adaptive Importance Sampling (PI-MAIS). Although PI-MAIS is parallelizable, in the iterative version of Table 7 the N independent processes cooperate together in Eq. (38) for providing unique global IS estimate. In the second adaptation scheme, we consider an interaction also at the upper level. Hence, we refer to this method as *Doubly Interacting Markov Adaptive Importance Sampling* (I^2-MAIS). In both cases, the corresponding technique provides an IS approximation of the target or, equivalently the estimate $\hat{I}_T$ and $\hat{Z}_T$ in Eq. (34), using NMT samples.

5.1.1. MCMC adaptation for PI-MAIS

The simplest option consists on applying one iteration of N parallel MCMC chains, one for each $\boldsymbol{\mu}_{n,t-1}$ returning $\boldsymbol{\mu}_{n,t}$, $n = 1, \ldots, N$. For instance, considering MH transitions, we have:

For $n = 1, \ldots, N$:

1. Draw $\boldsymbol{\mu}'$ from a proposal pdf $\varphi_n(\boldsymbol{\mu}|\boldsymbol{\mu}_{n,t-1}, \boldsymbol{\Lambda}_n)$.
2. Set $\boldsymbol{\mu}_{n,t} = \boldsymbol{\mu}'$ with probability
   $$\alpha = \min\left[1, \frac{\pi(\boldsymbol{\mu}')\varphi_n(\boldsymbol{\mu}_{n,t-1}|\boldsymbol{\mu}', \boldsymbol{\Lambda}_n)}{\pi(\boldsymbol{\mu}_{n,t-1})\varphi_n(\boldsymbol{\mu}'|\boldsymbol{\mu}_{n,t-1}, \boldsymbol{\Lambda}_n)}\right],$$
   otherwise set $\boldsymbol{\mu}_{n,t} = \boldsymbol{\mu}_{n,t-1}$ (with probability $1 - \alpha$).

Figure 3(a) illustrates this scenario. Each location parameter $\boldsymbol{\mu}_{n,t}$ is updated independently from the rest. Therefore, in PI-MAIS, the interaction among the different processes occurs only in the underlying IS layer of the hierarchical structure: the importance weights in Eq. (38) are built using the partial DM-MIS strategy with $\phi_t(\mathbf{x}) = \frac{1}{N}\sum_{n=1}^{N} q_{n,t}(\mathbf{x}|\boldsymbol{\mu}_{n,t}, \mathbf{C}_n)$. However, PI-MAIS can be parallelized if the IS steps are realized in a batch manner as explained in Section 4.2. Namely, all the weights can be computed and normalized at the end of the T iterations, after building the NT different proposals $q_{n,t}$. Note that the proposal pdfs φ_n can easily incorporate gradient information as in the Metropolis adjusted Langevin algorithm (MALA) [36, 23]. Different strategies for sharing information among the parallel chains can be also applied [13, 28, 29], having interaction at both levels: among the chains and within the IS estimation. Hence, in this case PI-MAIS becomes an I^2-MAIS scheme, detailed below.

Table 7. Population-Based Markov Adaptive Importance Sampling algorithms.

1. **Initialization:** Set $t = 1$, $\hat{I}_0 = 0$ and $H_0 = 0$. Choose the initial population

$$\mathcal{P}_0 = \{\boldsymbol{\mu}_{1,0}, ..., \boldsymbol{\mu}_{N,0}\},$$

and N covariance matrices $\mathbf{C}_n$ $(n = 1, \ldots, N)$. Choose also the parametric form of the N *normalized* proposal pdfs $q_{i,t}$ with parameter s $\boldsymbol{\mu}_{n,t}$ and $\mathbf{C}_n$. Let T be the total number of iterations.

2. For $t = 1, \ldots, T$:

   (a) **Update of the location parameters:** Perform one transition of one or more MCMC techniques over the current population,

$$\mathcal{P}_{t-1} = \{\boldsymbol{\mu}_{1,t-1}, ..., \boldsymbol{\mu}_{N,t-1}\},$$

   obtaining a new population,

$$\mathcal{P}_t = \{\boldsymbol{\mu}_{1,t}, ..., \boldsymbol{\mu}_{N,t}\}.$$

   (b) **IS steps:**

   (b-1) Draw $\mathbf{x}_{n,t}^{(m)} \sim q_{i,t}(\mathbf{x}|\boldsymbol{\mu}_{n,t}, \mathbf{C}_n)$ for $m = 1, \ldots, M$ and $n = 1, \ldots, N$.

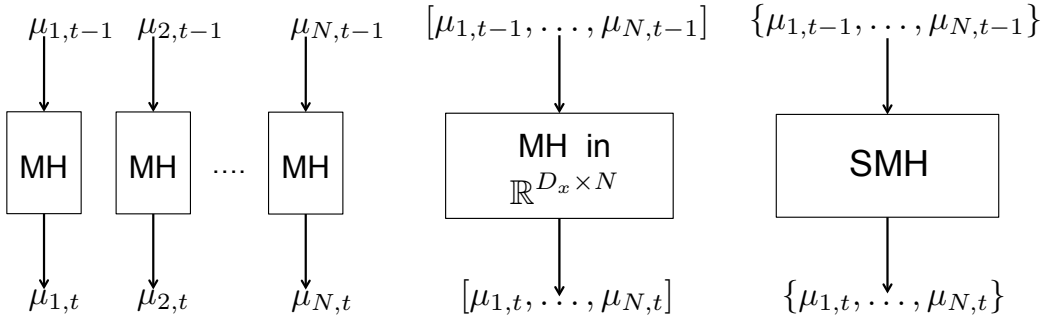
   (b-2) Compute the importance weights,

$$w_{n,t}^{(m)} = \frac{\pi(\mathbf{x}_{n,t}^{(m)})}{\frac{1}{N}\sum_{k=1}^{N} q_{k,t}(\mathbf{x}_{n,t}^{(m)}|\boldsymbol{\mu}_{k,t}, \mathbf{C}_k)}, \quad n = 1, \ldots, N, \quad m = 1, \ldots, M. \tag{38}$$

   (b-3) Set $S_t = \sum_{n=1}^{N}\sum_{m=1}^{M} w_{n,t}^{(m)}$, $H_t = H_{t-1} + S_t$, and normalize the weights

$$\bar{\rho}_{n,t}^{(m)} = \frac{w_{n,t}^{(m)}}{\sum_{\tau=1}^{t}\sum_{n=1}^{N}\sum_{m=1}^{M} w_t^{(m)}} = \bar{\rho}_{n,t-1}^{(m)}\frac{H_{t-1}}{H_t}.$$

   (c) **Outputs:** Return all the pairs $\{\mathbf{x}_\tau^{(m)}, \bar{\rho}_\tau^{(m)}\}$ for $m = 1, \ldots, M$ and $\tau = 1, \ldots, t$.



(a) Possible adapt. in PI-MAIS (b) Possible adapt. in I^2-MAIS (c) Possible adapt. in I^2-MAIS

Fig. 3. Different possible adaptation procedures for Population-based MAIS schemes. **(a)** One transition of N independent parallel MH chains ($\boldsymbol{\mu}_{n,t} \in \mathbb{R}^{D_x}$) for PI-MAIS. **(b)** One transition of an MH method working in the extended space $[\boldsymbol{\mu}_{1,t}, \ldots, \boldsymbol{\mu}_{N,t}] \in \mathbb{R}^{D_x \times N}$. **(c)** One transition of SMH [23, Chapter 5], considering the population of location parameter $\mathcal{P}_t = \{\boldsymbol{\mu}_{1,t}, ..., \boldsymbol{\mu}_{N,t}\}$.

5.1.2. MCMC adaptation for I^2-MAIS

In PI-MAIS the adaptation of the location parameters is performed independently from the rest of the population. Here, we discuss some non-independent strategies for updating the N location parameters $\boldsymbol{\mu}_{n,t}$. For this purpose, let us consider an extended state space $\mathbb{R}^{D_x \times N}$ and an extended target pdf

$$\bar{\pi}_g(\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_N) \propto \prod_{n=1}^{N} \pi(\boldsymbol{\mu}_n), \tag{39}$$

where each marginal $\pi(\boldsymbol{\mu}_n)$, $i = 1, ..., N$, coincides with the target pdf in Eq. (2). In this subsection, we describe two possible interacting adaptation procedures of the location parameters, which consider the generalized pdf in Eq. (39) as invariant density.

- *MH in the extended space* $\mathbb{R}^{D_x \times N}$*:* the simplest possibility is to apply a directly block-MCMC technique, transitioning from the matrix $\mathbf{P}_{t-1} = [\boldsymbol{\mu}_{1,t-1}, \ldots, \boldsymbol{\mu}_{N,t-1}]$ to the matrix $\mathbf{P}_t = [\boldsymbol{\mu}_{1,t}, \ldots, \boldsymbol{\mu}_{N,t}]$. Let us consider an MH method and a proposal pdf $\varphi(\mathbf{P}_t|\mathbf{P}_{t-1}) : \mathbb{R}^{D_x \times N} \to \mathbb{R}^{D_x \times N}$. For instance, one can consider a proposal of type

$$\varphi(\boldsymbol{\mu}_{1,t}, \ldots, \boldsymbol{\mu}_{N,t}|, \boldsymbol{\mu}_{1,t-1}, \ldots, \boldsymbol{\mu}_{N,t-1}) = \prod_{n=1}^{N} \varphi_n(\boldsymbol{\mu}_{n,t}|\boldsymbol{\mu}_{n,t-1}, \boldsymbol{\Lambda}_n).$$

  Thus, one transition is formed by the following steps:

  1. Draw $\mathbf{P}' \sim \varphi(\mathbf{P}|\mathbf{P}_{t-1})$, where $\mathbf{P}' = [\boldsymbol{\mu}'_1, \ldots, \boldsymbol{\mu}'_N]$.
  2. Set $\mathbf{P}_t = \mathbf{P}'$ with probability

$$\alpha = \min\left[1, \frac{\pi_g(\mathbf{P}')\varphi(\mathbf{P}_{t-1}|\mathbf{P}')}{\pi_g(\mathbf{P}_{t-1})\varphi(\mathbf{P}'|\mathbf{P}_{t-1})}\right],$$

     otherwise set $\mathbf{P}_t = \mathbf{P}_{t-1}$ (with probability $1 - \alpha$).

  At each iteration, N new samples $\boldsymbol{\mu}'_n$ are drawn (as in PI-MAIS) and therefore N new evaluations of π are required (i.e., one evaluation of π_g). When a new $\mathbf{P}$' is accepted, all the components of $\mathbf{P}_t$ differ from $\mathbf{P}_{t-1}$, unlike in the strategy described later. However, the probability of accepting a new population becomes dramatically lower as N grows.

- *Sample Metropolis-Hastings (SMH) algorithm* [23, Chapter 5]: SMH is an population-based MCMC technique, suitable for our purposes. At each iteration t, given the previous set

$$\mathcal{P}_{t-1} = \{\boldsymbol{\mu}_{1,t-1}, ..., \boldsymbol{\mu}_{N,t-1}\},$$

  a new possible parameter $\boldsymbol{\mu}_{0,t-1}$, drawn from an independent proposal $\varphi(\boldsymbol{\mu})$, is tested to be interchanged with another parameter in $\mathcal{P}_{t-1} = \{\boldsymbol{\mu}_{1,t-1}, ..., \boldsymbol{\mu}_{N,t-1}\}$. Namely, the underlying idea of SMH is to replace one "bad" sample in the population $\mathcal{P}_{t-1}$ with a possibly "better" one, according to a certain suitable probability α. The algorithm is designed so that, after a burn-in period, the elements in $\mathcal{P}_t$ are distributed according to $\pi_g(\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_N)$. Note that this means, fixing t, $\boldsymbol{\mu}_{1,t}, ..., \boldsymbol{\mu}_{N,t}$ are i.i.d. samples from $\pi(\mathbf{x})$. One iteration of SMH consists of the following steps:

  1. Draw a new candidate $\boldsymbol{\mu}_{0,t-1} \sim \varphi(\boldsymbol{\mu})$.
  2. Choose a "bad" sample, $\boldsymbol{\mu}_{k,t-1}$ with $k \in \{1, ..., N\}$, from the population according to a probability proportional to $\frac{\varphi(\boldsymbol{\mu}_{k,t-1})}{\pi(\boldsymbol{\mu}_{k,t-1})}$, which corresponds to the inverse of the importance sampling weights.
  3. Accept the new population, $\mathcal{P}_t = \{\boldsymbol{\mu}_{1,t} = \boldsymbol{\mu}_{1,t-1}, \ldots, \boldsymbol{\mu}_{k,t} = \boldsymbol{\mu}_{0,t-1}, \ldots, \boldsymbol{\mu}_{N,t} = \boldsymbol{\mu}_{N,t-1}\}$, with probability

$$\alpha(\mathcal{P}_{t-1}, \boldsymbol{\mu}_{0,t-1}) = \frac{\sum_{n=1}^{N} \frac{\varphi(\boldsymbol{\mu}_{n,t-1})}{\pi(\boldsymbol{\mu}_{n,t-1})}}{\sum_{i=0}^{N} \frac{\varphi(\boldsymbol{\mu}_{i,t-1})}{\pi(\boldsymbol{\mu}_{i,t-1})} - \min\limits_{0 \le i \le N} \frac{\varphi(\boldsymbol{\mu}_{i,t-1})}{\pi(\boldsymbol{\mu}_{i,t-1})}}. \tag{40}$$

     Otherwise, set $\mathcal{P}_t = \mathcal{P}_{t-1}$.

  Unlike in the previous strategy, the difference between $\mathcal{P}_{t-1}$ and $\mathcal{P}_t$ is at most one sample. Observe that α depends on $\mathcal{P}_{t-1}$ and the new possible parameter $\boldsymbol{\mu}_{0,t-1}$. However, at each iteration, only one new evaluation of π (and φ) is needed at $\boldsymbol{\mu}_{0,t-1}$, since the rest of the weights have already been computed in the previous steps (except for the initial iteration, where all need to be computed).

In both cases above, if φ are not properly chosen, the population of parameters hardly changes. However, the diversity in the population is preserved, unlike in the resampling procedure [8, 36]. The parameters of φ can be updated using an adaptive approach [20, 26] (see also Section 5.3). Techniques as the Normal Kernel Coupler [40] are other possible alternatives to the use of SMH.

5.2. Computational cost: comparison between PI-MAIS and I^2-MAIS

In the previously described algorithms, the samples generated by the Markov chain are not used for the estimation but only for updating the location parameters. The total number of samples involved in the final estimation is NMT, in both cases. The total number of evaluations of target E is bigger due to the MCMC implementation, i.e., $E > NMT$. Specifically, the total number of evaluations of the target is:

- $E = MNT + NT$, for PI-MAIS,
- $E = MNT + NT$, for I^2-MAIS with MH in the extended space $\mathbb{R}^{D_X \times N}$,
- $E = MNT + T$, for I^2-MAIS with SMH,

where we have taken into account that several evaluations of the target have been computed in the previous iterations. Moreover, the application of the MCMC techniques requires generation of L additional uniform random variables (r.v.'s) for performing the acceptance tests, and choosing a "bad" candidate in SMH, for instance. Specifically, we need: $L = NT$ uniform r.v.'s in the PI-MAIS, $L = T$ uniform r.v. for I^2-MAIS with MH in the extended space, $L = 2T$, one uniform r.v. and one multinomial r.v., for I^2-MAIS with SMH. However, the main computational effort is required for the target evaluation. It is important to remark that the computing time required in the multinomial sampling within SMH increases with N. Finally, we recall that, in the population-based MAIS schemes, we have considered a deterministic mixture procedure with $\Phi_{n,t}(\mathbf{x}) = \phi_t(\mathbf{x})$ which requires MN^2T evaluations of the proposal pdfs, $q_{n,t}(\mathbf{x})$, $n = 1, \ldots, N$ and $t = 1, \ldots, T$.

5.3. Adaptation of the covariance matrices $\mathbf{C}_n$

The adaptation of the scale parameters is in general a delicate task in adaptive Monte Carlo schemes. Indeed, a bad update of a scale parameter can easily jeopardize the rest of the adaptation and the global performance of the algorithm. Observe that a population-based method offers the opportunity of using jointly different scale parameters ($\mathbf{\Lambda}_1, \mathbf{\Lambda}_2, \ldots, \mathbf{\Lambda}_N$ and $\mathbf{C}_1, \mathbf{C}_2, \ldots, \mathbf{C}_N$, in this work), increasing the robustness of the resulting estimator. In order to design robust black-box methods, we advise against the adaptation of the scale parameters $\mathbf{\Lambda}_n$'s in the above level of the hierarchical procedure. We suggest a less sensitive approach in which the samples $\boldsymbol{\mu}_{n,t}$'s generated at the higher level can also be used for updating $\mathbf{C}_n$'s. For the sake of simplicity, let $\mathbf{C}_n$'s be covariance matrices. The covariance matrices $\mathbf{C}_n$'s could be adapted following different strategies proposed in literature. For instance, a possible procedure consists of setting

$$\mathbf{C}_{n,t} = \frac{1}{t}\sum_{\tau=1}^{t}(\boldsymbol{\mu}_{n,\tau} - \bar{\boldsymbol{\mu}}_{n,t})^\top(\boldsymbol{\mu}_{n,\tau} - \bar{\boldsymbol{\mu}}_{n,t}) + \beta_t\mathbf{C},$$

where $\bar{\boldsymbol{\mu}}_{n,t} = \frac{1}{t}\sum_{\tau=1}^{t}\boldsymbol{\mu}_{n,\tau}$. $\mathbf{C}$ is a covariance matrix chosen by the user and β_t is a value decreasing when t grows. An alternative is to consider a unique covariance matrix $\mathbf{C}_{n,t} = \mathbf{C}_t$ for all the proposals $q_{n,t}$ as suggested in [13], i.e.,

$$\mathbf{C}_t = \frac{1}{Nt}\sum_{n=1}^{N}\sum_{\tau=1}^{t}(\boldsymbol{\mu}_{n,\tau} - \bar{\bar{\boldsymbol{\mu}}}_t)^\top(\boldsymbol{\mu}_{n,\tau} - \bar{\bar{\boldsymbol{\mu}}}_t) + \beta_t\mathbf{C},$$

where $\bar{\bar{\boldsymbol{\mu}}}_t = \frac{1}{Nt}\sum_{n=1}^{N}\sum_{\tau=1}^{t}\boldsymbol{\mu}_{n,\tau}$.

6. NUMERICAL SIMULATIONS

In this section, we test the performance of the proposed scheme comparing them with other benchmark techniques. First of all, we tackle two challenging issues for adaptive Monte Carlo methods: multimodality in Section 6.1 and nonlinearity in Section 6.2. Furthermore, in Section 6.3 we consider an application of positioning and tuning model parameters in a wireless sensor network [1, 21, 32].

6.1. Multimodal target distribution

In this section, we test the novel proposed algorithms in a multimodal scenario, comparing with several other methods. Specifically, we consider a bivariate multimodal target pdf, which is itself a mixture of 5 Gaussians, i.e.,

$$\bar{\pi}(\mathbf{x}) = \frac{1}{5}\sum_{i=1}^{5}\mathcal{N}(\mathbf{x}; \nu_i, \mathbf{\Sigma}_i), \quad \mathbf{x} \in \mathbb{R}^2, \tag{41}$$

with means $\nu_1 = [-10, -10]^\top$, $\nu_2 = [0, 16]^\top$, $\nu_3 = [13, 8]^\top$, $\nu_4 = [-9, 7]^\top$, $\nu_5 = [14, -14]^\top$, and covariance matrices $\mathbf{\Sigma}_1 = [2,\ 0.6; 0.6,\ 1]$, $\mathbf{\Sigma}_2 = [2,\ -0.4; -0.4,\ 2]$, $\mathbf{\Sigma}_3 = [2,\ 0.8; 0.8,\ 2]$, $\mathbf{\Sigma}_4 = [3,\ 0; 0,\ 0.5]$ and $\mathbf{\Sigma}_5 = [2,\ -0.1; -0.1,\ 2]$. The main challenge in this example is the ability in discovering the 5 different modes of $\bar{\pi}(\mathbf{x}) \propto \pi(\mathbf{x})$. Since we know the moments of $\pi(\mathbf{x})$, we can easily assess the performance of the different techniques.

Given a random variable $\mathbf{X} \sim \bar{\pi}(\mathbf{x})$, we consider the problem of approximating via Monte Carlo the expected value $E[\mathbf{X}] = [1.6, 1.4]^\top$ and the normalizing constant $Z = 1$. Note that an adequate approximation of Z requires the ability of learning about all the 5 modes. We compare the performance in term of Mean Square Error (MSE) in the estimation using different sampling methodologies: **(a)** the AMIS technique [12], **(b)** three different PMC schemes[3], two of them proposed in [7, 8] and one PMC using a partial DM-MIS scheme with $\Phi_{n,t}(\mathbf{x}) = \phi_t(\mathbf{x})$, **(c)** N parallel independent MCMC chains and **(d)** the proposed PI-MAIS method. Moreover, we test two static MIS approaches, the standard MIS and a partial DM-MIS schemes with $\Phi_{n,t}(\mathbf{x}) = \phi_t(\mathbf{x})$, computing iteratively the final estimator.

For a fair comparison, all the techniques have been implemented in such a way that the number of total evaluations of the target density is $E = 2 \cdot 10^5$. All the involved proposal densities are Gaussian pdfs. More specifically, in PI-MAIS, we have set the following parameters: $N = 100$, $M \in \{1, 19, 99\}$, $T \in \{20, 100, 1000\}$ in order to fulfill $E = MNT + NT = (M + 1)NT = 2 \cdot 10^5$ (see Section 5.2). The proposal densities of the upper level of the hierarchical approach, $\varphi_n(\mathbf{x}|\boldsymbol{\mu}_{n,t}, \mathbf{\Lambda}_n)$, are Gaussian pdfs with covariance matrices $\mathbf{\Lambda}_n = \lambda^2 \mathbf{I}_2$ and $\lambda \in \{5, 10, 70\}$. The proposal densities used in the lower importance sampling level, $q_{n,t}(\mathbf{x}|\boldsymbol{\mu}_{n,t}, \mathbf{C}_n)$ are Gaussian pdfs with covariance matrices $\mathbf{C}_n = \sigma^2 \mathbf{I}_2$ and $\sigma \in \{0.5, 1, 2, 5, 10, 20, 70\}$. We also try different non-isotropic diagonal covariance matrices in both levels, i.e, $\mathbf{\Lambda}_n = \text{diag}(\lambda_{n,1}^2, \lambda_{n,2}^2)$, where $\lambda_{i,j} \sim \mathcal{U}([1, 10])$, and $\mathbf{C}_n = \text{diag}(\sigma_{n,1}^2, \sigma_{n,2}^2)$, where $\sigma_{n,j} \sim \mathcal{U}([1, 10])$ for $j \in \{1, 2\}$ and $n = 1, \dots N$. We test all these techniques using two different initializations: first, we choose deliberately a "bad" initialization of the initial location parameters, denoted as **In1**, in the sense that the initialization region does not contain the modes of π. Thus, we can test the robustness of the algorithms and their ability to improve the corresponding *static* approaches. Specifically, the initial location parameters are selected uniformly within the following square

$$\boldsymbol{\mu}_{n,0} \sim \mathcal{U}([-4, 4] \times [-4, 4]),$$

for $n = 1, \dots, N$. Different examples of this configuration are shown in Fig. 4 with squares. Secondly, we also consider a better initialization, denoted as **In2**, where the initialization region contains all the modes. Specifically, the initial location parameters are selected uniformly within the following square

$$\boldsymbol{\mu}_{n,0} \sim \mathcal{U}([-20, 20] \times [-20, 20]),$$

for $n = 1, \dots, N$. All the results are averaged over $2 \cdot 10^3$ independent experiments. Tables 8 and 9 show the Mean Square Error (MSE) in the estimation of the first component of $E[\mathbf{X}]$, with the initialization **In1** and **In2** respectively. Table 10 provides the MSE in the estimation of Z with **In1**. The best results in each column are highlighted in bold-face. In AMIS [12], the location and scale parameter of one proposal ($N = 1$) are adapted, using $\Phi_{1,t}(\mathbf{x}) = \xi_1(\mathbf{x})$ in the computation of the IS weights. Hence, in AMIS, we have tested different values of samples per iterations $M \in \{500, 10^3, 2 \cdot 10^3, 5 \cdot 10^3, 10^4\}$ and $T = \frac{E}{M}$. For the sake of simplicity, we directly show the worst and best results among the several simulations made with different parameters. PI-MAIS outperforms the other algorithms virtually for all the choices of the parameters, with both initializations. In general, a greater value of T is needed since the proposal pdfs are initially bad localized. Moreover, PI-MAIS always improves the performance of the static approaches. These two consideration show the benefit of the Markov adaptation. Hence, PI-MAIS presents more robustness with respect to the initial values and the choice of the scale parameters. Figure 4 depicts the initial (squares) and final (circles) configurations of the location parameters of the proposal densities for the standard PMC and the PI-MAIS methods, in a specific run and different values of $\sigma, \lambda \in \{3, 5\}$. In both cases, PI-MAIS guarantees a better covering of the modes of $\pi(\mathbf{x})$.

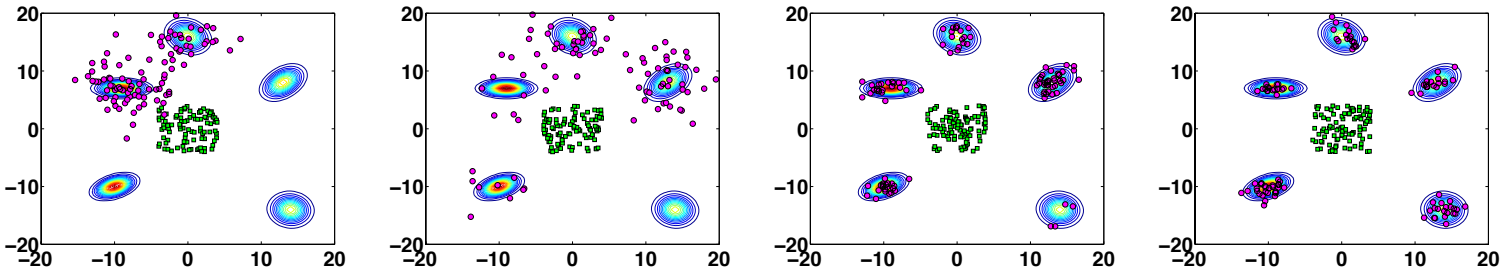
6.2. Nonlinear banana-shaped target distribution

Here we consider a bi-dimensional "banana-shaped" target distribution [20], which is a benchmark function in the literature due to its nonlinear nature. Mathematically, it is expressed as

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

where, we have set $B = 10$, $\eta_1 = 4$, $\eta_2 = 5$, and $\eta_3 = 5$. The goal is to estimate the expected value $E[X]$, where $X = [X_1, X_2] \sim p(x_1, x_2)$, by applying different Monte Carlo approximations. We approximately compute the true value $E[X] \approx$

[3] The standard PMC method [8] is described in Section 3.1.2.



(a) PMC ($N = 100, \sigma = 3$) (b) PMC ($N = 100, \sigma = 5$) (c) PI-MAIS ($N = 100, \lambda = 3$) (d) PI-MAIS ($N = 100, \lambda = 5$)

Fig. 4. Initial (squares) and final (circles) configurations of the location parameters of the proposal densities for the standard PMC and the PI-MAIS methods, in different specific runs. The initial configuration corresponds to **In1**.

$[-0.4845, 0]^\top$ using an exhaustive deterministic numerical method (with an extremely thin grid), in order to obtain the mean square error (MSE) of the following methods: standard PMC [8], the Mixture PMC [7], the AMIS [12], PI-MAIS and I^2-MAIS with SMH adaptation.

We consider Gaussian proposal distributions for all the algorithms. The initialization has been performed by randomly drawing the parameters of the Gaussians, with the mean of the n-th proposal given by $\boldsymbol{\mu}_{n,0} \sim \mathcal{U}([-6,-3] \times [-4,4])$ and its covariance matrix given by $\mathbf{C}_n = [\sigma_{n,1}^2 \quad 0;\ 0 \quad \sigma_{n,2}^2]^\top$. We have considered two cases: an *isotropic* setting where $\sigma_{n,1} = \sigma_{n,2} = \sigma \in \{1, 2, \ldots, 10\}$, and an *anisotropic* case with random selection of the parameters to test the robustness of the methods and where $\sigma_{n,1} \sim \mathcal{U}([1,20])$ and $\sigma_{n,2} \sim \mathcal{U}([1,20])$. Recall that in AMIS and Mixture PMC the covariance matrices are also adapted.

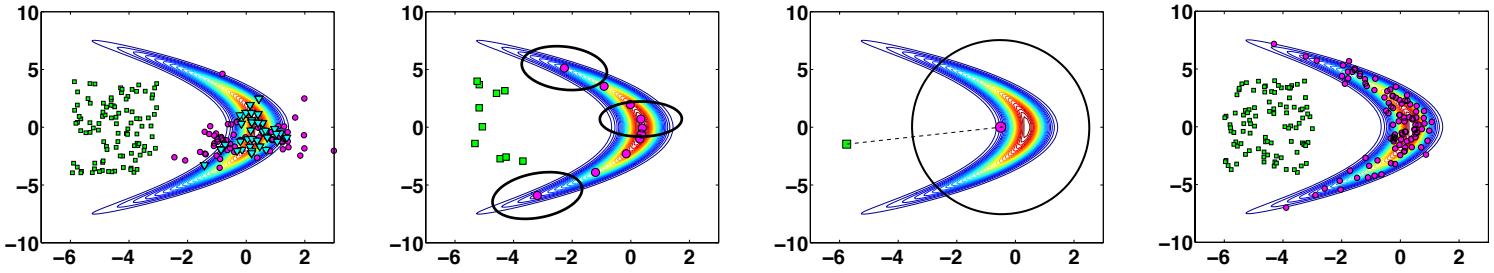
For each algorithm, we test several combinations of parameters, keeping fixed the total number of target evaluation, $E = 2 \cdot 10^5$. In the standard PMC method, described in Section 3.1.2), we consider $N \in \{50, 100, 10^3, 5 \cdot 10^3\}$ and $T = \frac{E}{N}$ (here $M = 1$). In Mixture PMC, we consider different number of component in the mixture proposal pdf $N \in \{10, 50, 100\}$, and different samples per proposal $S \in \{100, 200, 10^3, 2 \cdot 10^3, 5 \cdot 10^3\}$ at each iteration (here $T = \frac{E}{S}$). In AMIS, we test $S \in \{500, 10^3, 2 \cdot 10^3, 5 \cdot 10^3, 10^4\}$ and $T = \frac{E}{S}$ (we recall $N = 1$). The range of these values of parameters are chosen, after a preliminary study, in order to obtain the best performance from each technique. In PI-MAIS an I^2-MAIS, we set $N \in \{50, 100\}$. For the adaptation in PI-MAIS, we also consider Gaussian pdfs $\varphi_n(\mathbf{x}|\boldsymbol{\mu}_{n,t}, \boldsymbol{\Lambda}_n)$, covariance matrices $\boldsymbol{\Lambda}_n = \lambda^2 \mathbf{I}_2$ with $\lambda \in \{3, 5, 10, 20\}$. In I^2-MAIS, for the SMH method we use a Gaussian pdf with mean $[0,0]^\top$ and covariance matrix $\boldsymbol{\Lambda} = \lambda^2 \mathbf{I}_2$ and again $\lambda \in \{3, 5, 10, 20\}$. We test $M \in \{1, 9, 19\}$ for both, so that $T = \frac{E}{N(M+1)}$ for PI-MAIS and $T = \lfloor \frac{E}{NM+1} \rfloor$ for I^2-MAIS (see Section 5.2).

The results are averaged 500 over independent simulations, for each combination of parameters. Table 11 shows the smallest and highest MSE values obtained in the estimation of the expected value of the target, averaged between the two components of $E[X]$, achieved by the different methods. The smallest MSEs in each column (each σ) are highlighted with bold-faces. PI-MAIS and I^2-MAIS outperform the other techniques virtually for all the values of σ. In this example, AMIS also provides good results. Fig. 5 displays the initial (squares) and final (circles) configurations of the location parameters of the proposals for the different algorithms, in one specific run. Since in Mixture PMC and AMIS the covariance matrices are also adapted, we show the shape of some proposals as ellipses representing approximately 85% of probability mass. For, PMC we also depict a last resampling output with triangles, in order to show the loss in diversity. Unlike PMC, PI-MAIS ensures a better covering of the region of high probability.

6.3. Localization problem in a wireless sensor network

We consider the problem of positioning a target in a 2-dimensional space using range measurements. This problem appears frequently in localization applications in wireless sensor networks [1, 21, 32]. Namely, we consider a random vector $\mathbf{X} = [X_1, X_2]^\top$ to denote the target position in the plane $\mathbb{R}^2$. The position of the target is then a specific realization $\mathbf{X} = \mathbf{x}$. The range measurements are obtained from 3 sensors located at $\mathbf{h}_1 = [-10, 2]^\top$, $\mathbf{h}_2 = [8, 8]^\top$ and $\mathbf{h}_3 = [-20, -18]^\top$. The observation equations are given by

$$Y_j = a \log \left(\frac{||\mathbf{x} - \mathbf{h}_j||}{0.3} \right) + \Theta_j, \quad j = 1, \ldots, 3, \tag{42}$$



(a) PMC ($N = 100, \sigma = 1$) (b) Mixture PMC with 10 mixands ($\sigma = 5$) (c) AMIS ($\sigma = 5$) (d) PI-MAIS ($N = 100, \lambda = 3$)

Fig. 5. Initial (squares) and final (circles) configurations of the location parameters of the proposal densities for the banana-shaped target distribution, in one specific run and different methods. The Mixture PMC [7] and AMIS techniques [12] also adapt the covariance matrices (the ellipses show approximately 85% of the probability mass).

where Θ_j are independent Gaussian variables with identical pdfs, $\mathcal{N}(\vartheta_j; 0, \omega^2)$, $j = 1, 2$. We also consider a prior density over ω, i.e., $\Omega \sim p(\omega) = \mathcal{N}(\omega; 0, 25)I(\omega > 0)$, where $I(\omega > 0)$ is 1 if $\omega > 0$ and 0 otherwise. The parameter $A = a$ is also unknown and we again consider a Gaussian prior $A \sim p(a) = \mathcal{N}(a; 0, 25)$. Moreover, we also apply Gaussian priors over $\mathbf{X}$, i.e., $p(x_i) = \mathcal{N}(x_i; 0, 25)$ with $i = 1, 2$. Thus, the posterior pdf $\pi(x_1, x_2, a, \omega) = p(x_1, x_2, a, \omega|\mathbf{y})$ is

$$\pi(x_1, x_2, a, \omega) \propto \ell(\mathbf{y}|x_1, x_2, a, \omega)p(x_1)p(x_2)p(a)p(\omega),$$

where $\mathbf{y} \in \mathbb{R}^{D_y}$ is the vector of received measurements. We simulate $d = 30$ observations from the model ($D_y/3 = 10$ from each of the three sensors) fixing $x_1 = 3$, $x_2 = 3$, $a = -20$ and $\omega = 5$. With $D_y = 30$, the expected value of the target ($E[X_1] \approx 2.8749$, $E[X_2] \approx 3.0266$, $E[A] \approx 5.2344$, $E[\Omega] \approx 20.1582$)[4] is quite close to the true values.

Our goal is computing the expected value of $(X_1, X_2, A, \Omega) \sim \pi(x_1, x_2, a, \omega)$ via Monte Carlo, in order to provide an estimation of the position of the target, the parameter a and the standard deviation ω of the noise in the system. We apply PI-MAIS and three different PMC schemes (see example in Section 6.1, for a description), all using N Gaussian proposals. We initialize the cloud of location parameters spread out randomly in the space of the variables of interest, i.e.,

$$\boldsymbol{\mu}_{n,0} \sim \mathcal{N}(\boldsymbol{\mu}; \mathbf{0}, 30^2\mathbf{I}_4), \quad n = 1, ..., N,$$

and the scale parameters $\mathbf{C}_n = \text{diag}(\sigma^2_{n,1}, \ldots, \sigma^2_{n,4})\mathbf{I}_4$ with $n = 1, \ldots, N$. The values of the standard deviations $\sigma_{n,j}$ are chosen randomly for each Gaussian pdf. Specifically, $\sigma_{n,j} \sim \mathcal{U}([1, Q]),$, $j = 1, \ldots, 4$, where we have considered three possible values for Q, i.e., $Q \in \{5, 10, 30\}$. For the adaptation process of PI-MAIS, we consider also Gaussian proposals with covariance matrices $\boldsymbol{\Lambda}_n = \lambda^2\mathbf{I}_2$ and $\lambda \in \{5, 10, 70\}$. We also try different non-isotropic diagonal covariance matrices, i.e, $\boldsymbol{\Lambda}_n = \text{diag}(\lambda^2_{n,1}, \lambda^2_{i,2})$, where $\lambda_{n,j} \sim \mathcal{U}([1, 30])$.

For a fair comparison, all the techniques have been simulated with sets of parameters that yield the same number of target evaluations, fixed to $E = 2 \cdot 10^5$. In PI-MAIS, we have chosen parameters $N = 100$, $M = \{1, 19, 99\}$, $T = \{20, 100, 1000\}$. The PMC algorithms has been simulated with $N = 100$ and $T = 2000$. The MSE of the estimation (averaged over 3000 independent runs) are provided in Table 12. PI-MAIS outperforms always PMC when $\sigma_{n,j} \sim \mathcal{U}([1, 5])$ and $\sigma_{n,j} \sim \mathcal{U}([1, 10])$ whereas PMC provides better results for $\sigma_{n,j} \sim \mathcal{U}([1, 30])$. Therefore, the results show jointly the robustness and flexibility of the proposed PI-MAIS technique.

7. CONCLUSIONS

In this work, we have introduced a hierarchical generation procedure for designing adaptive Monte Carlo methods. We have described a hierarchical interpretation underlying two well-known techniques such us of a random walk proposal within an MH scheme and of the standard PMC method, thus pointing out the possible benefits of the layered strategy. Furthermore, we have applied this approach introducing a novel class of adaptive importance sampling (AIS) schemes. The novel class of AIS algorithms employs the determinist mixture (DM) idea [35, 38] for reducing the variance of the resulting IS estimators. We have extended the use of the DM strategy with respect to other techniques proposed in literature, providing a more general framework. From the estimation point of view, This framework includes different schemes proposed in literature [12, 30]

[4]These values have been obtained with a deterministic, expensive and exhaustive numerical integration method, using a thin grid.

as particular cases (although they differ for the employed adaptation procedure), and contains several others considering full or partial DM schemes. Advantages and computational cost have been discussed. Numerical comparisons with different benchmark techniques have been provided, confirming the benefit of the novel methodologies.

Acknowledgements

This work has been supported by the Spanish government's projects COMONSENS (CSD2008-00010), ALCIT (TEC2012-38800-C03-01), DISSECT (TEC2012-38058-C03-01), OTOSiS (TEC2013-41718-R), and COMPREHENSION (TEC2012-38883-C02-01), by the ERC grant 239784 and AoF grant 251170, and by the European Union 7th Framework Programme through the Marie Curie Initial Training Network "Machine Learning for Personalized Medicine" MLPM2012, Grant No. 316861.

References

[1] A. M. Ali, K. Yao, T. C. Collier, E. Taylor, D. Blumstein, and L. Girod. An empirical study of collaborative acoustic source localization. *Proc. Information Processing in Sensor Networks (IPSN07), Boston*, April 2007.

[2] C. Andrieu, N. de Freitas, A. Doucet, and M. Jordan. An introduction to MCMC for machine learning. *Machine Learning*, 50:5–43, 2003.

[3] F. Beaujean and Caldwell A. Initializing adaptive importance sampling with Markov chains. *arXiv:1304.7808*, 2013.

[4] Z. I. Botev and D. P. Kroese. An efficient algorithm for rare-event probability estimation, combinatorial optimization, and counting. *Methodology and Computing in Applied Probability*, 10(4):471–505, December 2008.

[5] Z. I. Botev, P. LEcuyer, and B. Tuffin. Markov chain importance sampling with applications to rare event probability estimation. *Statistics and Computing*, 23:271–285, 2013.

[6] A. Caldwell and C. Liu. Target density normalization for Markov Chain Monte Carlo algorithms. *arXiv:1410.7149*, 2014.

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

[8] O. Cappé, A. Guillin, J. M. Marin, and C. P. Robert. Population Monte Carlo. *Journal of Computational and Graphical Statistics*, 13(4):907–929, 2004.

[9] R. Casarin, R. V. Craiu, and F. Leisen. Interacting multiple try algorithms with different proposal distributions. *Statistics and Computing*, 23(2):185–200, 2013.

[10] S. Chib and I. Jeliazkov. Marginal likelihood from the metropolis-hastings output. *Journal of the American Statistical Association*, 96:270–281, 2001.

[11] N. Chopin. A sequential particle filter for static models. *Biometrika*, 89:539–552, 2002.

[12] J. M. Cornuet, J. M. Marin, A. Mira, and C. P. Robert. Adaptive multiple importance sampling. *Scandinavian Journal of Statistics*, 39(4):798–812, December 2012.

[13] R. Craiu, J. Rosenthal, and C. Yang. Learn from thy neighbor: Parallel-chain and regional adaptive MCMC. *Journal of the American Statistical Association*, 104(448):1454–1466, 2009.

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

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

[16] A. Doucet and X. Wang. Monte Carlo methods for signal processing. *IEEE Signal Processing Magazine*, 22(6):152–170, Nov. 2005.

[17] V. Elvira, L. Martino, D. Luengo, and M. Bugallo. Efficient multiple importance sampling estimators. *(to appear) IEEE Signal Processing Letters*, 2015.

[18] W. J. Fitzgerald. Markov chain Monte Carlo methods with applications to signal processing. *Signal Processing*, 81(1):3–18, January 2001.

[19] N. Friel and J. Wyse. Estimating the model evidence: a review. *arXiv:1111.1957*, 2011.

[20] H. Haario, E. Saksman, and J. Tamminen. An adaptive Metropolis algorithm. *Bernoulli*, 7(2):223–242, April 2001.

[21] 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*, 23(4):809–819, April 2005.

[22] J. Kotecha and Petar M. Djurić. Gibbs sampling approach for generation of truncated multivariate gaussian random variables. *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1999.

[23] F. Liang, C. Liu, and R. Caroll. *Advanced Markov Chain Monte Carlo Methods: Learning from Past Samples*. Wiley Series in Computational Statistics, England, 2010.

[24] R. Liesenfeld and J. F.Richard. Improving MCMC, using efficient importance sampling. *Computational Statistics and Data Analysis*, 53:272–288, 2008.

[25] J. S. Liu, F. Liang, and W. H. Wong. The multiple-try method and local optimization in metropolis sampling. *Journal of the American Statistical Association*, 95(449):121–134, March 2000.

[26] D. Luengo and L. Martino. Fully adaptive Gaussian mixture Metropolis-Hastings algorithm. *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2013.

[27] J. M. Marin, P. Pudlo, and M. Sedki. Consistency of the adaptive multiple importance sampling. *arXiv:1211.2548*, 2012.

[28] L. Martino, V. Elvira, D. Luengo, A. Artes, and J. Corander. Orthogonal MCMC algorithms. *IEEE Workshop on Statistical Signal Processing (SSP)*, pages 364–367, June 2014.

[29] L. Martino, V. Elvira, D. Luengo, A. Artes, and J. Corander. Smelly parallel MCMC chains. *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2015.

[30] L. Martino, V. Elvira, D. Luengo, and J. Corander. An adaptive population importance sampler: Learning from the uncertainty. *(to appear in) IEEE Transactions on Signal Processing; viXra.org:1405.0280*, 2015.

[31] L. Martino, V. Elvira, D. Luengo, and J. Corander. MCMC-driven adaptive multiple importance sampling. *Interdisciplinary Bayesian Statistics Springer Proceedings in Mathematics & Statistics (Chapter 8)*, 118:97–109, 2015.

[32] L. Martino and J. Míguez. A generalization of the adaptive rejection sampling algorithm. *Statistics and Computing*, 21(4):633–647, July 2011.

[33] E. F. Mendes, M. Scharth, and R. Kohn. Markov Interacting Importance Samplers. *arXiv:1502.07039*, 2015.

[34] R. Neal. MCMC using ensembles of states for problems with fast and slow variables such as Gaussian process regression. *arXiv:1101.0387*, 2011.

[35] A. Owen and Y. Zhou. Safe and effective importance sampling. *Journal of the American Statistical Association*, 95(449):135–143, 2000.

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

[37] J. Skilling. Nested sampling for general Bayesian computation. *Bayesian Analysis*, 1(4):833–860, June 2006.

[38] E. Veach and L. Guibas. Optimally combining sampling techniques for Monte Carlo rendering. *In SIGGRAPH 1995 Proceedings*, pages 419–428, 1995.

[39] X. Wang, R. Chen, and J. S. Liu. Monte Carlo Bayesian signal processing for wireless communications. *Journal of VLSI Signal Processing*, 30:89–105, 2002.

[40] G. R. Warnes. The Normal Kernel Coupler: An adaptive Markov Chain Monte Carlo method for efficiently sampling from multi-modal distributions. *Technical Report*, 2001.

[41] M. D. Weinberg. Computing the Bayes factor from a Markov chain Monte Carlo simulation of the posterior distribution. *arXiv:0911.1777*, 2010.

[42] X. Yuan, Z. Lu, and C. Z. Yue. A novel adaptive importance sampling algorithm based on Markov chain and low-discrepancy sequence. *Aerospace Science and Technology*, 29:253–261, 2013.

A. CONSISTENCY OF GAMIS ESTIMATORS

The consistency of the global estimators in Eq. (34) provided by GAMIS can be considered when number of samples per time step ($M \times N$) and the number of iterations of the algorithm (T) grow to infinity. For some exhaustive studies of specific cases, see the analysis in [36, 14] and [27]. Here we provide some brief arguments for explaining why $\hat{I}_T$ and $\hat{Z}_T$ obtained by a GAMIS scheme are, in general, consistent.[5] Let us assume that $q_{n,t}$'s have heavier tails than $\bar{\pi}(\mathbf{x}) \propto \pi(\mathbf{x})$. Note that the global estimator $\hat{I}_T$ can be seen as a result of a static batch MIS estimator involving L different mixture-proposals $\Phi_{n,t}(\mathbf{x})$ and $J = NMT$ total number of samples. The weights $w_{n,t}^{(m)}$ built using $\Phi_{n,t}(\mathbf{x})$ in the denominator of the IS ratio are suitable importance weights yielding consistent estimators, as explained in detail in Appendices B-C. Hence, for a finite number of iterations $T < \infty$, when $M \to \infty$ (or $N \to \infty$), the consistency can be guaranteed by standard IS arguments, since it is well known that $\hat{Z}_T \to Z$ and $\hat{I}_T \to I$ as $M \to \infty$, or $N \to \infty$ [36].

Furthermore, for $T \to \infty$ and $N, M < \infty$ finite, we have a convex combination, given in Eq. (36), of conditionally independent (consistent but biased) IS estimators [36]. Indeed, although in an adaptive scheme the proposals depend on the previous configurations of the population, the samples drawn at each iteration are conditionally independent of the previous ones, and independent of each other drawn at the same iteration. The bias is due to unknown Z (see Eq. (4)), and hat $\hat{Z}_T$ is used to replace Z. However, $\hat{Z}_T \to Z$ as $T \to \infty$, as discussed in [36, Chapter 14]: hence, $\hat{I}_T$ is asymptotically unbiased as $T \to \infty$. Furthermore, in this work, we consider an adaption procedure completely independent of the estimation steps and, as a consequence, independent of the samples $\mathbf{x}$ included in the estimation procedure.

B. DRAWING FROM A MIXTURE OF PDFS

Let us consider a mixture composed of N normalized densities with equal weights, i.e.,

$$\psi(\mathbf{x}) = \frac{1}{N} \sum_{n=1}^{N} q_n(\mathbf{x}). \tag{43}$$

The classical procedure for drawing $J = NM$ samples from $\psi(\mathbf{x})$ is (starting with $j = 1$):

1. Draw $k^* \in \{1, \ldots, N\}$ with equal weights $\frac{1}{N}$.
2. Draw $\mathbf{x}^{(j)} \sim q_{k^*}(\mathbf{x})$.
3. Set $j = j + 1$ and repeat until $j > J = NM$.

In this way, each sample $\mathbf{x}^{(j)}$ is distributed according $\psi(\mathbf{x})$ and, as a consequence, the entire set,

$$\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(J)}\} \sim \psi(\mathbf{x}),$$

is also distributed as $\psi(\mathbf{x})$. An alternative procedure, more deterministic than the previous one, consists of the following steps (starting with $n = 1$):

1. Draw M independent samples from $q_n(\mathbf{x})$, i.e., $\mathbf{x}_n^{(1)}, \ldots, \mathbf{x}_n^{(M)} \sim q_n(\mathbf{x})$.
2. Set $n = n + 1$ and repeat until $n > N$.

[5] A complete analysis should take in account the chosen adaptive procedure since, in general, the adaptation uses the information of previous weighted samples. However, in this work we consider an adaption procedure completely independent of the estimation steps, as clarified in the next section.

In this case, we have $\mathbf{x}_n^{(m)} \sim q_n(\mathbf{x})$ for $n = 1, \ldots, N$ and $m = 1, \ldots, M$, *but* the joint set

$$\{\mathbf{x}_1^{(1)}, \ldots, \mathbf{x}_1^{(M)}, \ldots, \mathbf{x}_N^{(1)}, \ldots, \mathbf{x}_N^{(M)}\} \sim \psi(\mathbf{x}),$$

is again distributed as $\psi(\mathbf{x})$ (recall that $J = NM$). This alternative approach can be interpreted as an application of a variance reduction method [36] for sampling a mixture. Moreover for the same arguments, we can also assert that, given R indices $j_r \in \{1, \ldots, N\}$ with $r = 1, \ldots, R$, we have

$$\{\mathbf{x}^{(j_1)}, \ldots, \mathbf{x}^{(j_R)}\} \sim \frac{1}{R} q_{j_1}(\mathbf{x}) + \ldots + \frac{1}{R} q_{j_R}(\mathbf{x}).$$

This is the theoretical base of the partial DM-MIS procedure described in Section 4.

C. IS APPROACHES USING WITH MULTIPLE PROPOSAL PDFS

Recall that our goal is computing efficiently the integral $I = \frac{1}{Z}\int_{\mathcal{X}} f(\mathbf{x})\pi(\mathbf{x})d\mathbf{x}$ where f is a generic smooth function and $Z = \int_{\mathcal{X}} \pi(\mathbf{x})d\mathbf{x} < \infty$ with $\pi(\mathbf{x}) \geq 0$ for all $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^{D_x}$. Let us assume that we have to normalized proposal pdfs, $q_1(\mathbf{x})$ and $q_2(\mathbf{x})$, from which we intend to draw M_1 and M_2 samples respectively:

$$\mathbf{x}_1^{(1)}, \ldots, \mathbf{x}_1^{(M_1)} \sim q_1(\mathbf{x}) \qquad \text{and} \qquad \mathbf{x}_2^{(1)}, \ldots, \mathbf{x}_2^{(M_2)} \sim q_2(\mathbf{x}).$$

There are at least two procedures to build a joint IS estimator: the standard multiple importance sampling (MIS) approach and the full deterministic mixture (DM-MIS) scheme.

C.1. Standard IS approach

The simplest approach [36, Chapter 14] is computing the classical IS weights:

$$w_1^{(i)} = \frac{\pi(\mathbf{x}_1^{(i)})}{q_1(\mathbf{x}_1^{(i)})}, \quad w_2^{(k)} = \frac{\pi(\mathbf{x}_2^{(k)})}{q_2(\mathbf{x}_2^{(k)})}, \tag{44}$$

with $i = 1, \ldots, M_1$ and $k = 1, \ldots, M_2$. The IS estimator is then built by normalizing them jointly, i.e., computing

$$\hat{I}_{IS} = \frac{1}{S_{tot}} \left(\sum_{i=1}^{M_1} w_1^{(i)} f(\mathbf{x}_1^{(i)}) + \sum_{k=1}^{M_2} w_2^{(k)} f(\mathbf{x}_2^{(k)}) \right), \tag{45}$$

where $S_{tot} = \sum_{i=1}^{M_1} w_1^{(i)} + \sum_{k=1}^{M_2} w_2^{(k)}$. Note that, by defining $S_{tot} = S_1 + S_2$, where the two partial sums are given by $S_1 = \sum_{i=1}^{M_1} w_1^{(i)}$ and $S_2 = \sum_{k=1}^{M_2} w_2^{(k)}$, and considering the normalized weights, $\bar{w}_1^{(i)} = \frac{w_1^{(i)}}{S_1}$ and $\bar{w}_2^{(k)} = \frac{w_2^{(k)}}{S_2}$, Eq. (45) can be rewritten as

$$\hat{I}_{IS} = \frac{1}{S_1 + S_2} \left(S_1 \hat{I}_1 + S_2 \hat{I}_2 \right) = \frac{S_1}{S_1 + S_2} \hat{I}_1 + \frac{S_2}{S_1 + S_2} \hat{I}_2, \tag{46}$$

where $\hat{I}_1$ and $\hat{I}_2$ are the two partial IS estimators, obtained by considering only one proposal pdf. This procedure can be easily extended for $N > 2$ different proposal pdfs, obtaining the complete IS estimator as the convex combination of the N partial IS estimators:

$$\hat{I}_{IS} = \frac{\sum_{n=1}^{N} S_n \hat{I}_n}{\sum_{n=1}^{N} S_n}, \tag{47}$$

where $\mathbf{x}_n^{(1)}, \ldots, \mathbf{x}_n^{(M_n)} \sim q_n(\mathbf{x})$, $w_n^{(i)} = \pi(\mathbf{x}_n^{(i)})/q_n(\mathbf{x}_n^{(i)})$, $S_n = \sum_{i=1}^{M_n} w_n^{(i)}$ and $\hat{I}_n = \sum_{i=1}^{M_n} w_n^{(i)} f(\mathbf{x}_n^{(i)})$.

C.2. Deterministic mixture

An alternative approach is based on the deterministic mixture sampling idea [35, 38] described previously in Appendix B. Considering $N = 2$ proposals q_1, q_2, and setting

$$\mathcal{Z} = \left\{ \mathbf{x}_1^{(1)}, \ldots, \mathbf{x}_1^{(M_1)}, \mathbf{x}_2^{(1)}, \ldots, \mathbf{x}_2^{(M_2)} \right\},$$

with $\mathbf{x}_n^{(m_n)} \in \mathbb{R}^{D_x}$ ($n \in \{1,2\}$ and $1 \leq m_n \leq M_n$), the weights are now defined as

$$w_n^{(m_n)} = \frac{\pi(\mathbf{x}_n^{(m_n)})}{\frac{M_1}{M_1+M_2} q_1(\mathbf{x}_n^{(m_n)}) + \frac{M_2}{M_1+M_2} q_2(\mathbf{x}_n^{(m_n)})}. \tag{48}$$

In this case, the *complete* proposal is considered to be a mixture of q_1 and q_2, weighted according to the number of samples drawn from each one. Note that, unlike in the standard procedure for sampling from a mixture, a deterministic and fixed number of samples are drawn from each proposal in the DM approach. It can be easily shown that the set $\mathcal{Z}$ of samples drawn in this deterministic way is distributed according to the mixture $q(\mathbf{z}) = \frac{M_1}{M_1+M_2} q_1(\mathbf{z}) + \frac{M_2}{M_1+M_2} q_2(\mathbf{z})$ [35]. The DM estimator is finally given by

$$\hat{I}_{DM} = \frac{1}{S_{tot}} \sum_{n=1}^{2} \sum_{m_n=1}^{M_n} w_n^{(m_n)} f(\mathbf{x}_n^{(m_n)}),$$

where $S_{tot} = \sum_{n=1}^{2} \sum_{m_n=1}^{M_n} w_n^{(m_n)}$ and the $w_n^{(m_n)}$ are given by (48). For $N > 2$ proposal pdfs, the DM estimator can also be easily generalized:

$$\hat{I}_{DM} = \frac{1}{\sum_{n=1}^{N} \sum_{m_n=1}^{M_n} w_n^{(m_n)}} \sum_{n=1}^{N} \sum_{m_n=1}^{M_n} w_n^{(m_n)} f(\mathbf{x}_n^{(m_n)}),$$

with

$$w_i = \frac{\pi(\mathbf{x}_n^{(m_n)}}{\sum_{n=1}^{N} \frac{M_n}{M_{tot}} q_n(\mathbf{x}_n^{(m_n)})},$$

and $M_{tot} = M_1 + M_2 + \ldots + M_N$. On the one hand, the DM approach is more stable than the IS method, thus providing a better performance in terms of a reduced variance of the corresponding estimator, as shown in the following section. On the other hand, it needs to evaluate every proposal M_{tot} times instead of only M_n times (in the standard MIS procedure), and therefore is more costly from a computational point of view. However, this increased computational cost is negligible when the proposal is much cheaper to evaluate than the target, as it often happens in practical applications.

D. DISTRIBUTION OF THE LOCATION PARAMETERS AFTER RESAMPLING

Let us consider a standard PMC scheme [8], with N different proposal pdfs $q_1, \ldots, q_N$. Recall that in a standard PMC method, one sample ($M = 1$) is drawn from each proposal. For the sake of simplicity, since we are considering a generic iteration t, let us simplify the notation denoting as $\mathbf{x}_i = \mathbf{x}_{n,t} \sim q_{n,t}(\mathbf{x}|\boldsymbol{\mu}_{n,t}, \mathbf{C}_n)$ ($1 \leq n \leq N, 1 \leq t \leq T$), and as $q_n(\mathbf{x}) = q_{n,t}(\mathbf{x}|\boldsymbol{\mu}_{n,t}, \mathbf{C}_n)$. Moreover, we define as

$$\mathbf{m}_{\neg n} = [\mathbf{x}_1, \ldots, \mathbf{x}_{n-1}, \mathbf{x}_{n+1}, \ldots, \mathbf{x}_N]^\top,$$

the vector containing all the samples except for the n-th. Let us also denote as $\mathbf{x}' \in \{\mathbf{x}_1 \ldots, \mathbf{x}_N\}$, a generic location parameter after applying a multinomial resampling step with replacement. Hence, the distribution of $\mathbf{x}$ is given by

$$\phi(\mathbf{x}') = \int_{\mathcal{X}^N} \hat{\pi}^{(N)}(\mathbf{x}') \left[\prod_{n=1}^{N} q_n(\mathbf{x}_n)\right] d\mathbf{x}_1 \ldots d\mathbf{x}_N, \tag{49}$$

where $\hat{\pi}^{(N)}(\mathbf{x}') = \sum_{n=1}^{N} \bar{w}_{n,t} \delta(\mathbf{x}' - \mathbf{x}_i)$, and $\bar{w}_{i,t} = \frac{w_{i,t}}{\sum_{n=1}^{N} w_{n,t}}$, with $i = 1, \ldots, N$. Then, after some straightforward rearrangements, Eq. (50) can be rewritten as

$$\phi(\mathbf{x}') = \sum_{j=1}^{N} \left(\int_{\mathcal{X}^{N-1}} \frac{\frac{\pi(\mathbf{x}_j)}{q_j(\mathbf{x}_j)}}{\sum_{n=1}^{N} \frac{\pi(\mathbf{x}_n)}{q_n(\mathbf{x}_n)}} \left[\prod_{\substack{n=1 \\ n \neq j}}^{N} q_n(\mathbf{x}_n)\right] d\mathbf{m}_{\neg j} \right) \delta(\mathbf{x}' - \mathbf{x}_j). \tag{50}$$

Finally, we can write

$$\phi(\mathbf{x}') = \pi(\mathbf{x}') \sum_{j=1}^{N} \left(\int_{\mathcal{X}^{N-1}} \frac{1}{N\hat{Z}} \left[\prod_{\substack{n=1 \\ n \neq j}}^{N} q_n(\mathbf{x}_n)\right] d\mathbf{m}_{\neg j} \right), \tag{51}$$

where $\hat{Z} = \frac{1}{N} \sum_{n=1}^{N} \frac{\pi(\mathbf{x}_n)}{q_n(\mathbf{x}_n)}$ is the estimate of the normalizing constant of the target obtained using the classical IS weights. When $N \to \infty$, then $\hat{Z} \to Z$ [36], and thus $\phi(\mathbf{x}) \to \frac{1}{Z}\pi(\mathbf{x}) = \bar{\pi}(\mathbf{x})$.

ALGORITHM			$\sigma = 0.5$	$\sigma = 1$	$\sigma = 2$	$\sigma = 5$	$\sigma = 10$	$\sigma = 70$	$\sigma_{n,j} \sim \mathcal{U}([1, 10])$
PI-MAIS ($N = 100$)	$\lambda = 5$	$M = 99, T = 20$	1.2760	0.5219	0.5930	0.0214	0.0139	0.1815	0.0107
		$M = 19, T = 100$	0.2361	0.1205	0.0422	0.0087	0.0140	0.1868	0.0052
		$M = 1, T = 1000$	0.1719	0.0019	0.0155	0.0103	0.0273	0.3737	0.0070
	$\lambda = 10$	$M = 99, T = 20$	1.0195	0.1546	0.2876	0.0178	0.0133	0.1789	0.0098
		$M = 19, T = 100$	0.1750	0.0120	0.0528	0.0086	0.0136	0.1856	0.0050
		$M = 1, T = 1000$	0.1550	**0.0021**	**0.0020**	0.0095	0.0252	0.3648	0.0066
	$\lambda = 70$	$M = 99, T = 20$	16.9913	5.5790	1.4925	0.0382	0.0128	0.1834	0.0252
		$M = 19, T = 100$	2.6693	0.9182	0.1312	0.0147	0.0143	0.1844	0.0120
		$M = 1, T = 1000$	0.3014	0.1042	0.0136	0.0115	0.0267	0.3697	0.0093
	$\lambda_{n,j} \sim \mathcal{U}([1, 10])$	$M = 99, T = 20$	1.0707	0.5364	0.3523	0.0199	0.0121	0.1919	0.0094
		$M = 19, T = 100$	0.2481	0.0595	0.1376	**0.0075**	0.0144	0.1899	**0.0049**
		$M = 1, T = 1000$	**0.1046**	0.0037	0.0045	0.0099	0.0274	0.3563	0.0065
STATIC STANDARD MIS	$\Phi_{n,t}(\mathbf{x}) = q_{n,t}(\mathbf{x})$		29.56	41.95	64.51	2.17	0.0147	0.1914	4.55
STATIC PARTIAL DM-MIS	$\Phi_{n,t}(\mathbf{x}) = \phi_t(\mathbf{x})$		29.28	47.74	75.22	0.2424	0.0124	0.1789	0.0651
AMIS [12]	(best results)		124.22	121.21	100.23	0.8640	**0.0121**	**0.0136**	0.7328
	(worst results)		125.43	123.38	114.82	16.92	0.0128	18.66	13.49
PMC [8]	$N = 100, T = 2000$		112.99	114.11	47.97	2.34	0.0559	2.41	0.3017
PMC WITH PARTIAL DM-MIS			111.92	107.58	26.86	0.6731	0.0744	2.42	0.0700
MIXTURE PMC [7]			110.17	113.11	50.23	2.75	0.0521	2.57	0.6194
PARALLEL INDEP. MH CHAINS	$N = 100, T = 2000$		1.6910	1.7640	1.8832	1.4133	0.2969	0.5475	7.3446

Table 8. **(Ex-Sect 6.1)** MSE of the estimation of the expected value (first component) with the initialization **In1**. For all the techniques, the total number of evaluations of the target is $E = 2 \cdot 10^5$. We recall that, in AMIS [12], $N = 1$ and $\Phi_{1,t}(\mathbf{x}) = \xi_1(\mathbf{x})$. The last row corresponds to the application of $N = 100$ parallel MH methods (the same used in PI-MAIS for the adaptation) where the random walk proposals have covariance matrices $\mathbf{C} = \sigma^2 \mathbf{I}_2$. The lengths of the chains, as well as of the PMC runs, is $T = 2000$ for maintaining the same number of target evaluations E than in PI-MAIS. For the techniques which adapt the covariance matrices of the proposal pdfs, the values of σ have been employed as initial values of the scale parameters. For AMIS, we show the best and worst results obtained testing different combinations of $M \in \{500, 10^3, 2 \cdot 10^3, 5 \cdot 10^3, 10^4\}$ and $T = \frac{E}{M}$. The best results, in each column, are highlighted with bold-faces.

ALGORITHM			$\sigma = 0.5$	$\sigma = 1$	$\sigma = 2$	$\sigma = 5$	$\sigma = 10$	$\sigma = 70$	$\sigma_{i,j} \sim \mathcal{U}([1, 10])$
PI-MAIS ($N = 100$)	$\lambda = 5$	$M = 99, T = 20$	0.6096	0.0657	0.0023	0.0056	**0.0124**	0.1768	0.0051
		$M = 19, T = 100$	0.2878	0.0358	**0.0010**	0.0050	0.0127	0.1802	0.0038
		$M = 1, T = 1000$	0.1244	**0.0011**	0.0014	0.0091	0.0242	0.3510	0.0064
	$\lambda = 10$	$M = 99, T = 20$	0.9236	0.0543	0.0021	0.0062	0.0137	0.1815	0.0054
		$M = 19, T = 100$	0.2294	0.0077	0.0012	0.0054	0.0132	0.1890	0.0044
		$M = 1, T = 1000$	0.0786	0.0042	0.0014	0.0086	0.0256	0.3503	0.0066
	$\lambda = 70$	$M = 99, T = 20$	5.9889	0.3662	0.0082	0.0089	0.0140	0.1841	0.0093
		$M = 19, T = 100$	1.6670	0.0871	0.0045	0.0080	0.0139	0.1971	0.0074
		$M = 1, T = 1000$	0.2579	0.0134	0.0024	0.0097	0.0258	0.3543	0.0082
	$\lambda_{i,j} \sim \mathcal{U}([1, 10])$	$M = 99, T = 20$	0.5623	0.0417	0.0025	0.0059	**0.0124**	0.1848	0.0056
		$M = 19, T = 100$	0.2704	0.0204	0.0011	**0.0048**	0.0136	0.1726	**0.0037**
		$M = 1, T = 1000$	**0.0750**	0.0014	0.0013	0.0089	0.0247	0.3540	0.0066
STATIC STANDARD MIS	$\Phi_{n,t}(\mathbf{x}) = q_{n,t}(\mathbf{x})$		12.00	9.40	10.26	7.67	0.5443	0.1764	4.37
STATIC PARTIAL DM-MIS	$\Phi_{n,t}(\mathbf{x}) = \phi_t(\mathbf{x})$		10.14	0.9469	0.0139	0.0100	0.0146	0.1756	0.0106
AMIS [12]	(best results)		113.97	112.70	107.85	44.93	0.7404	**0.0141**	31.02
	(worst results)		116.66	115.62	111.83	70.62	9.43	18.62	58.63
PMC [8]	$N = 100, T = 2000$		111.54	110.78	90.21	2.29	0.0631	2.42	0.3082
PMC WITH PARTIAL DM-MIS			23.16	7.43	7.56	0.6420	0.0720	2.37	0.0695
MIXTURE PMC [7]			25.43	10.68	6.29	0.6142	0.0727	2.55	0.1681
PARALLEL INDEP. MH CHAINS	$N = 100, T = 2000$		1.3813	1.3657	1.2942	1.0178	0.3644	1.0405	5.3211

Table 9. **(Ex-Sect 6.1)** MSE of the estimation of the expected value (first component). For all the techniques, the total number of evaluations of the target is again $E = 2 \cdot 10^5$. In this case, we have applied the initialization **In2**, differently from Table 8. The best results, in each column, are highlighted with bold-faces.

ALGORITHM			$\sigma = 0.5$	$\sigma = 1$	$\sigma = 2$	$\sigma = 5$	$\sigma = 10$	$\sigma = 70$	$\sigma_{n,j} \sim \mathcal{U}([1, 10])$
PI-MAIS ($N = 100$)	$\lambda = 5$	$M = 99, T = 20$	0.0388	0.0120	0.0070	0.0002	0.0001	0.0016	0.0001
		$M = 19, T = 100$	0.0031	0.0013	0.0004	0.0001	0.0001	0.0017	0.0001
		$M = 1, T = 1000$	0.0016	0.0001	0.0001	0.0001	0.0002	0.0031	0.0001
	$\lambda = 10$	$M = 99, T = 20$	0.0217	0.0046	0.0040	0.0001	0.0001	0.0016	0.0002
		$M = 19, T = 100$	0.0019	0.0002	0.0005	0.0001	0.0001	0.0017	0.0001
		$M = 1, T = 1000$	0.0016	0.0001	0.0001	**$8 \cdot 10^{-5}$**	0.0002	0.0031	0.0001
	$\lambda = 70$	$M = 99, T = 20$	6.3732	0.2713	0.0226	0.0003	0.0001	0.0016	0.0002
		$M = 19, T = 100$	0.1082	0.0114	0.0019	0.0001	0.0001	0.0017	0.0001
		$M = 1, T = 1000$	0.0038	0.0009	0.0001	0.0001	0.0002	0.0033	0.0001
	$\lambda_{n,j} \sim \mathcal{U}([1, 10])$	$M = 99, T = 20$	0.0350	0.0101	0.0043	0.0001	0.0001	0.0015	0.0001
		$M = 19, T = 100$	0.0029	0.0007	0.0010	**$8 \cdot 10^{-5}$**	$9 \cdot 10^{-5}$	0.0017	**$9 \cdot 10^{-5}$**
		$M = 1, T = 1000$	0.0014	0.0001	**$9 \cdot 10^{-5}$**	0.0001	0.0002	0.0036	0.0001
STATIC STANDARD MIS	$\Phi_{n,t}(\mathbf{x}) = q_{n,t}(\mathbf{x})$		$3.94 \cdot 10^{4}$	$7.12 \cdot 10^{7}$	$1.07 \cdot 10^{3}$	0.0113	0.0001	0.0016	0.2190
STATIC PARTIAL DM-MIS	$\Phi_{n,t}(\mathbf{x}) = \phi_t(\mathbf{x})$		$9.51 \cdot 10^{8}$	$4.60 \cdot 10^{5}$	15.34	0.0016	0.0001	0.0016	0.0005
AMIS [12]	(best results)		15.92	15.66	12.81	0.0069	**$8 \cdot 10^{-5}$**	**0.0001**	0.0002
	(worst results)		15.97	15.92	14.87	0.4559	0.0001	1.62	0.0084
PMC [8]	$N = 100, T = 2000$		33.53	17.10	14.42	0.4249	0.0015	0.0016	0.3542
PMC WITH PARTIAL DM-MIS			15.85	14.31	1.81	0.0402	0.0002	0.0016	0.0004
MIXTURE PMC [7]			14.51	12.09	3.56	0.0287	0.0002	0.0015	0.0010

Table 10. **(Ex-Sect 6.1)** MSE of the estimation of the normalizing constant Z with the initialization **In1**. For all the techniques, the total number of evaluations of the target is $E = 2 \cdot 10^5$. The smallest MSE for each σ is bold-faced.

ALGORITHM		$\sigma = 0.5$	$\sigma = 1$	$\sigma = 2$	$\sigma = 3$	$\sigma = 5$	$\sigma = 10$	$\sigma = 70$	$\sigma_{i,j} \sim \mathcal{U}([1, 20])$
PI-MAIS	Worst	0.0083	0.0081	0.0012	0.0005	0.0050	0.0126	0.1126	0.0218
	Best	0.0025	**0.0001**	**0.0002**	**0.0001**	0.0002	0.0003	0.0361	0.0004
I^2-MAIS	Worst	0.0335	0.0227	0.0053	0.0044	0.0041	0.0096	0.2130	0.0181
	Best	0.0082	0.0025	0.0013	0.0008	**0.0001**	**0.0002**	0.0265	**0.0003**
PMC [8]	Worst	0.0670	0.0461	0.0209	0.0093	0.0055	0.0072	9.4749	0.1065
	Best	0.0210	0.0164	0.0069	0.0016	0.0015	0.0011	0.0262	0.0026
MIXTURE PMC [7]	Worst	3.5772	0.0113	0.0044	0.0066	0.0174	0.0267	0.0913	0.0103
	Best	0.0092	0.0020	0.0018	0.0035	0.0034	0.0055	0.0138	0.0025
AMIS [12]	Worst	0.0040	0.0039	0.0040	0.0016	0.0011	0.0012	0.0035	0.0013
	Best	**0.0023**	0.0028	0.0023	0.0009	0.0003	0.0004	**0.0023**	0.0007

Table 11. **(Ex-Section-6.2)** Bi-dimensional banana-shaped distribution example: Best and worst results in terms of MSE, obtained with the different techniques for different values of σ. The smallest MSE for each σ is bold-faced.

ALGORITHM			$\sigma_{i,j} \sim \mathcal{U}([1, 5])$	$\sigma_{i,j} \sim \mathcal{U}([1, 10])$	$\sigma_{i,j} \sim \mathcal{U}([1, 30])$
PI-MAIS	$\lambda = 5$	$M = 99, T = 20$	0.3819	0.3508	0.3626
		$M = 19, T = 100$	0.0728	0.0738	0.0710
		$M = 1, T = 1000$	0.0173	**0.0164**	0.0171
	$\lambda = 10$	$M = 99, T = 20$	0.5701	0.5943	0.5605
		$M = 19, T = 100$	0.1389	0.1429	0.1425
		$M = 1, T = 1000$	0.0401	0.0408	0.0393
	$\lambda_{i,j} \sim \mathcal{U}([1, 30])$	$M = 99, T = 20$	0.3758	0.3795	0.4028
		$M = 19, T = 100$	0.0741	0.0793	0.0771
		$M = 1, T = 1000$	**0.0169**	0.0167	**0.0162**
PMC [8]	$N = 100, T = 2000$		0.0642	0.4345	0.1533
PMC WITH PARTIAL DM-MIS			0.0524	0.3163	0.0817
MIXTURE PMC [7]			0.0577	0.2870	0.4083

Table 12. **(Ex-Sect 6.3)** MSE of the estimation of $E[(X_1, X_2, A, \Omega)]$ using different techniques, keeping constant the total number of target evaluation, $E = 2\ 10^5$. The best results, in each column, are highlighted with bold-faces.