Hybrid Population Monte Carlo

Ali Mousavi and Víctor Elvira, *Senior Member, IEEE*

Abstract—Importance sampling (IS) is a powerful Monte Carlo (MC) technique for approximating intractable integrals, for instance in Bayesian inference. The performance of IS relies heavily on the appropriate choice of the so-called proposal distribution. Adaptive IS (AIS) methods iteratively improve target estimates by adapting the proposal distribution. Recent AIS research focuses on enhancing proposal adaptation for high-dimensional problems, while addressing the challenge of multi-modal targets. In this paper, a new class of AIS methods is presented, utilizing a hybrid approach that incorporates weighted samples and proposal distributions to enhance performance. This approach belongs to the family of population Monte Carlo (PMC) algorithms, where a population of proposals is adapted to better approximate the target distribution. The proposed hybrid population Monte Carlo (HPMC) implements a novel two-step adaptation mechanism. In the first step, a hybrid method is used to generate the population of the preliminary proposal locations based on both weighted samples and location parameters. We use Hamiltonian Monte Carlo (HMC) to generate the preliminary proposal locations. HMC has a good exploratory behavior, especially in high dimension scenarios. In the second step, the novel cooperation algorithms are performing to find the final proposals for the next iteration. HPMC achieves a significant performance improvement in high-dimensional problems when compared to the state-of-the-art algorithms. We discuss the statistical properties of HPMC and show its high performance in two challenging benchmarks.

Index Terms—Adaptive importance sampling, Monte Carlo, Statistical signal processing

I. INTRODUCTION

IN the field of statistical signal processing, many tasks require the computation of expectations with respect to a probability density function (PDF). Unfortunately, deriving analytical solutions for these expectations is often unfeasible in many real-world challenging problems. A relevant example occurs in the realm of Bayesian or probabilistic inference. Here, the objective is to formulate a posterior distribution involving unknown parameters. This distribution is constructed by combining data, through a likelihood model, and previous information represented by the prior distribution. However, a persistent challenge arises—the posterior distribution is frequently beyond reach, primarily due to the complexity of computing the marginal likelihood. This complexity poses significant hurdles in numerous practical scenarios. Various approximation techniques are available to tackle this issue, the most popular of which is the Monte Carlo (MC) methodology [1]–[3].

MC methods have demonstrated their efficiency in addressing this problem by generating a collection of random samples. These samples serve the purpose of approximating a target probability distribution and integrals related to it. Importance sampling (IS) is a popular class of MC methods. It is theoretically sound, simple-to-understand, and widely applicable. The basic IS involves: a) generating samples by simulating from another distribution, called proposal, (b) weighting each sample based on the discrepancy between the target and the proposal distribution, and (c) approximating integrals or density using weighted samples.. The efficiency of the IS estimators is driven by the choice of the proposal distribution, which is a hard and very relevant problem. This choice is even more challenging in situations where we only have access to the evaluation of an unnormalized version of the posterior distribution, as in the case of Bayesian inference. One significant methodological advancement to address the limitations of IS is multiple importance sampling (MIS), where samples are generated from multiple proposal distributions. A comprehensive MIS framework was recently introduced in [4], demonstrating various sampling and weighting schemes that can be utilized. Despite the benefits of MIS, adapting the proposals can be even more advantageous, especially in high-dimensional scenarios. The intuition behind adaptive IS (AIS) methods is to adapt one or a mixture of proposals, iteratively improving the quality of the estimator by better fitting the proposals [5].

AIS methods gained prominence in MC computations following the introduction of the population Monte Carlo (PMC) [6] family of algorithms. PMC uses a resampling step based on previous weighted samples to adapt a population of proposal PDFs [7], [8]. Several AIS algorithms have been proposed since the publication of PMC. We refer the reader to [9], [10] for an exhaustive review. AIS methods are diverse and widely used, with key differences in how proposals are selected, weighted, and adapted. A critical step is proposal adaptation, which can be categorized into three main approaches. The first category adapts proposals using the latest samples, as in PMC methods [6], with enhancements such as N-PMC [11], M-PMC [12], and DM-PMC [13] and more advanced methods based on local curvature of the target like [14], [15]. The second category utilizes all past samples, as in AMIS [16] and APIS [17], or sequential Monte Carlo methods like those in [18], [19]. The third category employs hierarchical adaptations, such as GAPIS [20], ULA-based methods [21], [22], and advanced layered AIS methods like LAIS [23], MCIS [24], and HAIS [25]. The proposed method in this paper belongs to the PMC family, leveraging a hybrid adaptation approach from both the first and third categories.

One challenging issue in layered AIS algorithms is the lack of sufficient cooperation during the adaptation procedure to generate proposal parameters for the next iteration. Additionally, recent layered methods often require an external

A. Mousavi is with Department of Computer Engineering, Neyshabur Branch, Islamic Azad University, Neyshabur, Iran (e-mail: mousavi@iau-neyshabur.ac.ir).

V. Elvira is with the School of Mathematics, University of Edinburgh, United Kingdom (e-mail: victor.elvira@ed.ac.uk).

proposal PDF that is unrelated to the target PDF. For instance, the PI-MAIS and I^2-MAIS algorithms in LAIS use an independent traditional Metropolis-Hastings (MH) algorithm for adaptation. HAIS, as a notable extension of LAIS, introduces a two-layered AIS that retains advantageous features from both Hamiltonian Monte Carlo (HMC) [26], [27] and AIS. HAIS employs a more efficient adaptive procedure, offering additional benefits such as parallelization and cooperation via resampling for efficient inference. A major limitation of the adaptation procedures in layered methods is their exclusive dependence on samples or proposal parameters generated within the current iteration. Specifically, the adaptation and sampling steps in these AIS methods are decoupled, meaning the samples generated for estimation are not utilized in the adaptation process. Another significant challenge in layered AIS algorithms is the lack of sufficient cooperation during the adaptation phase to generate proposal parameters for subsequent iterations. Additionally, recent layered methods often require an external proposal PDF that is unrelated to the target PDF.

In this paper, we present hybrid-PMC (HPMC), a new class of AIS methods that employs a hybrid adaptation strategy, leveraging both weighted samples generated from the proposals and the proposal parameters (see Fig. 1-d). This method extends the PMC framework by enhancing the adaptation step to achieve an effective balance between local and global exploration, making it particularly suitable for high-dimensional, multi-modal targets. Moreover, it retains the advantageous exploratory behavior of HMC, especially in high-dimensional scenarios. The proposed algorithm introduces a two-step adaptation mechanism. In the first step, preliminary parameters (locations) from the proposals are generated based on two sets: *i)* *weighted samples generated from the proposals*: These enrich the diversity in the population of proposal locations, increasing the likelihood of discovering local features of the target; *ii)* *location parameter of the proposals*: These guide the proposals toward unexplored regions, promoting global exploration. By leveraging both weighted samples and location parameters, our approach reduces the number of iterations needed to reach the desired estimate, leading to faster convergence.

Another significant contribution of this work is the novel cooperation step, where information among preliminary locations is exchanged to redirect the lost proposals toward the optimal path based on the target PDF. Two novel methods are proposed for determining the final set of proposal locations for the next iteration of the AIS procedure: cooperation by resampling and cooperation by weighted mixture modeling. This cooperation step introduces a theoretically sound and consistent procedure to enhance global exploration. Notably, this cooperation step does not require any free parameter tuning. Finally, HPMC implements the DM weighting technique for both target estimation and adaptation, which has demonstrated superior performance compared to other methods in the AIS literature.

In a nutshell, the contributions of this paper are as follows:
- We propose a new class of AIS methods that, for the first time, combines weighted samples and location parameters to achieve an effective balance between local and global

exploration, significantly accelerating estimator convergence.
- A novel cooperation step is introduced to redirect lost proposals, enhancing global exploration and optimizing the allocation of the computational complexity without requiring any parameter tuning.
- HPMC incorporates the DM weighting technique, which demonstrates superior performance in both target estimation and proposal adaptation compared to existing techniques

The rest of the paper is organized as follows. Section II, describes the background in Monte Carlo. The novel HMPC method is presented in Sections III-A to C, and its theoretical foundation is discussed in Section III-D. The paper concludes with three numerical examples and a discussion on computational complexity comparison in Section IV. Finally, some concluding remarks is explained in Section V.

II. BACKGROUND IN MONTE CARLO

In the Bayesian framework, the variable of interest $\mathbf{x} \in \mathbb{R}^{d_x}$ with related measurements $\mathbf{y} \in \mathbb{R}^{d_y}$, is characterized through the posterior probability function or the target probability density function (PDF) known as the

$$\widetilde{\pi}(\mathbf{x}|\mathbf{y}) = \frac{\ell(\mathbf{y}|\mathbf{x})p_0(\mathbf{x})}{Z(\mathbf{y})} \propto \pi(\mathbf{x}) = \ell(\mathbf{y}|\mathbf{x})p_0(\mathbf{x}), \quad (1)$$

where $\ell(\mathbf{y}|\mathbf{x})$ is the likelihood function, $p_0(\mathbf{x})$, represents the prior PDF, and $Z(\mathbf{y})$ is the model evidence. We alleviate the notation by dropping $\mathbf{y}$, since our goal is to address more general frameworks beyond the Bayesian inference.

In many applications the goal is to compute the integral

$$I = \int_{\mathbb{R}^{d_x}} f(\mathbf{x})\widetilde{\pi}(\mathbf{x})d\mathbf{x} = \frac{1}{Z} \int_{\mathbb{R}^{d_x}} f(\mathbf{x})\pi(\mathbf{x})d\mathbf{x}, \quad (2)$$

where $f : \mathbb{R} \to \mathbb{R}^{d_x}$ is any integrable function w.r.t. $\widetilde{\pi}(\mathbf{x})$ and $Z = \int \pi(\mathbf{x})d\mathbf{x}$. In many cases of interest, it is impossible or too complicated to find an analytical solution to this integral, in which case we resort to the Monte Carlo (MC) methods.

A. Importance sampling

Importance sampling (IS) is a MC technique. To comprehend IS, we start by rewriting Eq. (2) as

$$I = \mathbb{E}_{\widetilde{\pi}(\mathbf{x})}[f(\mathbf{X})] = \mathbb{E}_{q(\mathbf{x})}\left[f(\mathbf{X})\frac{\widetilde{\pi}(\mathbf{X})}{q(\mathbf{X})}\right], \quad (3)$$

where $\mathbb{E}(.)$ denotes expectation for $\mathbf{X} \sim q(\mathbf{x})$ and $q(\mathbf{x})$ is the so-called proposal distribution with the constraint that $q(\mathbf{x}) > 0$ for all $\mathbf{x}$ where $\widetilde{\pi}(\mathbf{x})f(\mathbf{x}) \neq 0$. The basic idea of IS is to approximate this integral by generating M random samples $\mathbf{x}_m$ from a simple proposal PDF $q(\mathbf{x})$, and each sample is associated with an importance weight w_m. The IS procedure involves two fundamental steps:

1) Sampling: M iid samples are generated as

$$\mathbf{x}_m \sim q(\mathbf{x}), \ m = 1, \cdots, M.$$

2) Weighting: each sample $\mathbf{x}_m$ is assigned an importance weight as

$$w_m = \frac{\pi(\mathbf{x}_m)}{q(\mathbf{x}_m)}.$$

The resulting pairs of $\{\mathbf{x}_m, w_m\}_{m=1}^M$ are used to approximate the integral I w.r.t. to the target distribution $\widetilde{\pi}(\mathbf{x})$ as

$$\widehat{I}_{\text{IS}}(\mathbf{x}) = \frac{1}{MZ} \sum_{m=1}^M w_m f(\mathbf{x}_m). \qquad (4)$$

Thus we can construct the *unnormalized IS* (UIS) estimator $\widehat{I}_{\text{IS}}$, which is an unbiased and consistent estimator of I. When Z is not available, the *self-normalized IS* (SNIS) estimator can be constructed by substituting the unbiased estimator $\widehat{Z} = \frac{1}{M} \sum_{m=1}^M w_m$ for Z in Eq. (4), i.e.,

$$\widetilde{I}_{\text{IS}}(\mathbf{x}) = \frac{1}{M\widehat{Z}} \sum_{m=1}^M w_m f(\mathbf{x}_m) = \sum_{m=1}^M \bar{w}_m f(\mathbf{x}_m). \qquad (5)$$

SNIS is a consistent but biased estimator and as M increases, with the bias decreasing more rapidly than the variance as M increases.

B. Multiple importance sampling

The IS method has been extended to scenarios where samples are drawn from multiple proposals, $q_n(\mathbf{x})_{n=1}^N$, rather than a single one. This approach, called multiple importance sampling (MIS), introduced by Veach and Guibas [28], is particularly effective for fitting multi-modal target distributions. In standard MIS, we generate K samples from each proposal, i.e.,

$$\mathbf{x}_{n,k} \sim q_n(\mathbf{x}), \ n = 1, ..., N, \ and \ k = 1, ..., K.$$

Then, the generalized MIS estimator as in [28] is given by

$$\widetilde{I}_{\text{MIS}}(\mathbf{x}) = \sum_{n=1}^N \frac{1}{K} \sum_{k=1}^K g_n(\mathbf{x}_{n,k}) \frac{f(\mathbf{x}_{n,k})\pi(\mathbf{x}_{n,k})}{q_n(\mathbf{x}_{n,k})}, \qquad (6)$$

where $g_n(\mathbf{x}) \geq 0$ is a weighting function corresponding to the n-th proposal which satisfies $\sum_{n=1}^N g_n(\mathbf{x}) = 1$ for all $\mathbf{x}$. The most common function for $g_n(\mathbf{x})$ is the *balance heuristic* with

$$\begin{aligned} g_n(\mathbf{x}) &= \frac{Kq_n(\mathbf{x})}{\sum_{j=1}^N Kq_j(\mathbf{x})} \\ &= \frac{1}{N} \frac{q_n(\mathbf{x})}{\sum_{j=1}^N q_j(\mathbf{x})}. \end{aligned} \qquad (7)$$

In this case, the MIS estimator $\widetilde{I}_{\text{MIS}}(\mathbf{x})$ becomes the balance heuristic estimator, i.e.,

$$\widetilde{I}_{\text{MIS}}(\mathbf{x}) = \frac{1}{ZN} \sum_{n=1}^N \frac{1}{K} \sum_{k=1}^K f(\mathbf{x}_{n,k}) w_{n,k}, \qquad (8)$$

with $w_{n,k} = \pi(\mathbf{x}_{n,k})/\frac{1}{N} \sum_{j=1}^N q_j(\mathbf{x}_{n,k})$. We can also refer to Eq. (8) as the deterministic mixture MIS (DM-MIS) estimator [29]. In the standard MIS weighting scheme, we define

$$w_{n,k} = \frac{\pi(\mathbf{x}_{n,k})}{q_n(\mathbf{x}_{n,k})}, \ n = 1, ..., N, \ and \ k = 1, ..., K, \qquad (9)$$

which corresponds to the standard MIS (s-MIS) estimator [6]. It is possible to construct a MIS estimator using either of the weighting functions. However, the MIS estimator with DM weights outperforms the estimator with standard weights. Specifically, it has been shown that $\text{Var}[\widetilde{I}_{\text{DM-MIS}}(\mathbf{x})] \leq \text{Var}[\widetilde{I}_{\text{s-MIS}}(\mathbf{x})]$ [4]. Note that both weighting schemes involve the same number of target evaluations; however, the DM-MIS estimator requires more computational effort in terms of proposal evaluations (with N^2 evaluations needed for the DM weights compared to N evaluations for the standard weights). Several efficient methods have been proposed in the literature to reduce computational complexity [30], [31]. There are various weighting functions in the literature. For further details on these weighting schemes, which lead to consistent estimators, we refer the reader to [4].

C. Adaptive importance sampling

Adaptive importance sampling (AIS) iteratively adapts one or several proposals to approximate the target PDF. AIS methods typically involve three key steps:

1) Sampling: Generate a set of samples from one or multiple proposal PDF(s).

2) Weighting: Use a weighting function to compute the weight of each sample obtained in the previous step.

3) Adaptation: Update the proposal(s) to improve the sampling for the next iteration.

A typical self-normalized AIS estimator to approximate the integral I w.r.t. the target PDF $\widetilde{\pi}$ up to iteration t is given by

$$\widetilde{I}_{\text{AIS}}(\mathbf{x}) = \frac{1}{KNt\widehat{Z}} \sum_{\tau=1}^t \sum_{n=1}^N \sum_{k=1}^K w_{n,k}^{(\tau)} f(\mathbf{x}_{n,k}^{(\tau)}), \qquad (10)$$

and

$$\widehat{Z} = \frac{1}{KNt} \sum_{\tau=1}^t \sum_{n=1}^N \sum_{k=1}^K w_{n,k}^{(\tau)}, \qquad (11)$$

where $\mathbf{x}_{n,k}^{(\tau)}$ with $w_{n,k}^{(\tau)}$ is the k-th weighted sample generated from n-th proposal PDF at iteration τ of the AIS algorithm.

AIS methods vary in proposal types, weighting schemes, and adaptation strategies, with proposal adaptation being crucial. The most significant adaptation procedures in AIS families can be classified into three distinct categories. Fig. 1(a-c) graphically illustrates the three categories, with each subplot depicting the dependency of proposal parameter adaptation and the sample generation process. The index t represents the iteration number, n denotes the proposal number, and $\theta_n^{(t)}$ are the parameters of the proposals to be adapted. For simplicity of presentation, we assume that only one sample is drawn from the n-th proposal (i.e., $K = 1$). Each red arrow specifies what is used to adapt a proposal. The descriptions and algorithms from the literature for each category are as follows:

(1) The proposal parameters are adapted using the last set of drawn samples (Fig. 1-a). The population Monte Carlo (PMC) methods, as introduced in [6], belong to this category. A key characteristic of PMC methods is their use of resampling techniques to adapt the location parameters of the proposals [8]. Other approaches, such as N-PMC [11] with nonlinear weighting and M-PMC [12] employing stochastic

expectation-maximization, have enhanced the performance of the standard PMC method. Further advancements in weighting and resampling techniques are presented in [13], [32], which introduces the DM-PMC, GR-PMC, LR-PMC, and Diverse-PMC methods. Some methods [14], [15] have further improved PMC samplers for high-dimensional, multimodal target distributions by incorporating the target's local curvature after resampling. However, these strategies are inefficient in high dimensions as they require inverting the Hessian matrix of the target distribution.

(2) The proposal parameters are adapted using all drawn samples up to the latest iteration (Fig. 1-b). Examples from this category include AMIS [16], which introduces a temporal estimator based on moment matching, and APIS [17], which utilizes multiple proposals across epochs. Two other works [18], [19] on sequential MC samplers (SMC) also construct a more diverse and rich sample set of the target posterior by leveraging samples from all iterations.

(3) Some recent methods aim at adapting the proposals through an independent process from the past samples (Fig. 1-c). This class of algorithms could be called layered or hierarchical methods for the adaptation in IS. In the upper layer, proposal parameters are adapted while in the lower layer weighted samples are generated to estimate the target. As an example, GAPIS [20] exploits the gradient and the Hessian of the target in the upper layer to adapt the location and covariance parameters, respectively. Other methods, such as [21], [22], adapt the location parameters by performing several steps of the unadjusted Langevin algorithm (ULA) at each sample. More advanced structures, such as LAIS [23], [33], MCIS [24], and HAIS [25], have incorporated Markov chain Monte Carlo (MCMC) mechanisms for the adaptation of proposals. The method we propose in this paper belongs to the PMC framework family, utilizing a hybrid adaptation from both the first and third categories (Fig. 1-d).

III. HYBRID POPULATION MONTE CARLO

A. Summary of the algorithm

In this section, we present the novel *hybrid population Monte Carlo* (HPMC) method. HPMC is an adaptive importance sampling (AIS) method that uses multiple proposals to generate samples at each iteration, that belongs to the family of PMC algorithms. The algorithm is described in Alg. 1. The procedure starts by initializing N (parametric) proposals, $q_n(\mathbf{x}; \boldsymbol{\mu}_n^{(t)}, \boldsymbol{\nu}_n)$ where $\boldsymbol{\mu}_n^{(1)} \in \mathbb{R}^{d_x}$ is the initial location parameter (e.g., the mean vector in a Gaussian PDF) and $\boldsymbol{\nu}_n$ contains the other parameters (e.g., $\boldsymbol{\nu}_n = \boldsymbol{\Sigma}_n$ is the covariance matrix in a Gaussian distribution). The algorithm runs for T iterations, involving three main steps: sampling (Step 1), weighting (Step 2), and adaptation (Step 3). In Step 1 (sampling), a set of multiple proposals is used to draw samples. In Step 2 (weighting), the normalized IS weight is calculated for each sample. Finally, in Step 3 (adaptation), the parameters of all proposals are updated for the next iteration. The adaptation process is a hybrid approach that utilizes all weighted samples generated from the proposals as well as the location parameter of the proposals. The adaptation step

(a) The proposal parameters are adapted using the last set of drawn samples.

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

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

(d) The proposal parameters are adapted using both the generated weighted samples and the location parameters of the proposals.
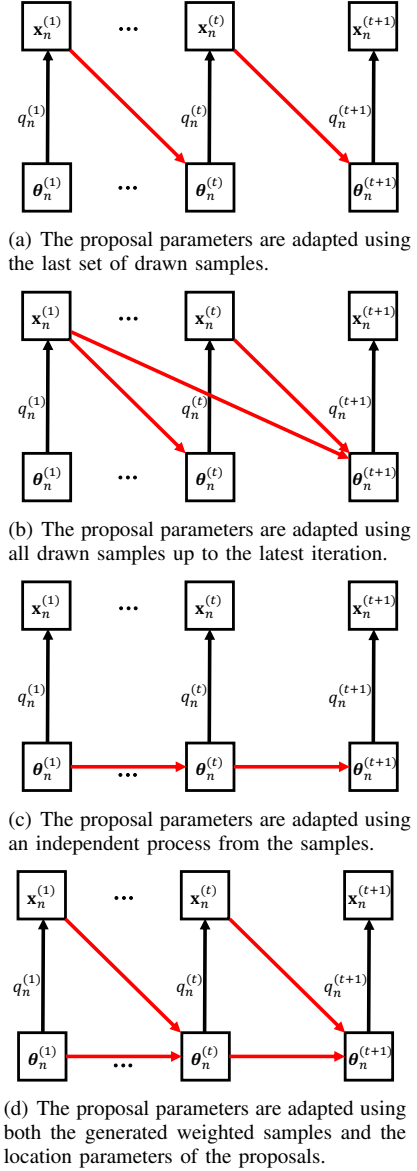
Fig. 1: Graphical description of the three main existing classes of AIS algorithms (a-c), and the new class (d).

follows of a two-step procedure. In Step 1(a), we generate a population of preliminary locations of the proposals based on: i) weighted samples generated from the proposals and ii) location parameters. In the subsequent cooperation step (Step 2(b)), information from all preliminary locations is exchanged to generate new final proposal locations for the next iteration. Finally, the algorithm outputs the set of weighted samples to estimate $\widetilde{I}$ and $\widehat{Z}$. In the following, we discuss the steps in detail.

B. Sampling and weighting

In the sampling step, at each iteration t, exactly K samples are simulated from each of the N proposals, resulting in a total of KN samples. The weighting operation is applied to compute the importance weights using the deterministic mixture (DM) scheme according to Eq. (13) and then normal-

---

Algorithm 1: Hybrid population Monte Carlo

Input: initial location, $\{\boldsymbol{\mu}_n^{(1)}\}_{n=1}^N$, and scale, $\{\boldsymbol{\nu}_n\}_{n=1}^N$, of proposals, N proposals, K sample per proposal, T iterations

for $t = 1, ..., T$ **do**

    Step 1 (sampling): draw K samples per individual proposal or mixand,

$$\mathbf{x}_{n,k}^{(t)} \sim q_n(\mathbf{x}; \boldsymbol{\mu}_n^{(t)}, \boldsymbol{\nu}_n), \; n = 1, ..., N, \; and \; k = 1, ..., K. \quad (12)$$

    Step 2: (Weighting): compute the deterministic mixture (DM) weight of each sample by

$$w_{n,k}^{(t)} = \frac{\pi(\mathbf{x}_{n,k}^{(t)})}{\sum_{i=1}^N q_i(\mathbf{x}_{i,k}^{(t)}; \boldsymbol{\mu}_i^{(t)}, \boldsymbol{\nu}_i)}, \quad (13)$$

    and normalize through

$$\bar{w}_{n,k}^{(t)} = \frac{w_{n,k}^{(t)}}{\sum_{i=1}^N \sum_{j=1}^K w_{i,j}^{(t)}}. \quad (14)$$

    Step 3: (adaptation): update location of proposals for the next iteration:

    3a) **Preliminary locations generation**: obtain new population of proposal locations by two separate operations:

        i) generate N preliminary locations, P, by applying N parallel LR sampling with the weights from Eq. (15).

        ii) generate N preliminary locations, Q, by applying one iteration of N parallel HMC blocks.

    3b) **Final locations generation through cooperation**: generate final population of locations, $\{\boldsymbol{\mu}_1^{(t+1)}, \boldsymbol{\mu}_2^{(t+1)}, ..., \boldsymbol{\mu}_N^{(t+1)}\}$, for the next iteration according to the methods in Section III-C (Step 3(b))

end

Output: a set of KNT samples with associated weights, $\{\mathbf{x}_{n,k}^{(t)}, w_{n,k}^{(t)}\}$ to obtain the estimation.

---

ized through Eq. (14). DM weighting has shown to provide an estimator with lower variance compared to the others (as discussed in Section II-B). Each of these operations is represented as a separate module in Fig. 2, illustrating the step-by-step structure of the algorithm.

C. Adaptation process

The proposed adaptation process is a hybrid approach that includes two steps: Step 3(a), preliminary location generation, and Step 3(b), final location generation through cooperation. Below, we provide a detailed description of this novel approach:

Step 3(a): Preliminary locations generation

We introduce two distinct operations to find two sets of preliminary locations of the proposal PDFs, $\{\boldsymbol{\mu}_1^{(t+1)*}, \boldsymbol{\mu}_2^{(t+1)*}, \cdots, \boldsymbol{\mu}_N^{(t+1)*}\}$, discussed as follows.

i) Preliminary locations generation based on weighted samples

We propose using the current set of NK samples from Eq. (12), weighted by Eq. (13), to generate the population of N preliminary locations for the next iteration (as illustrated in the top-right block of the adaptation module in Fig. 2). Next, N local resampling (LR) [13] procedures are independently performed in parallel within each subset of K samples. The n-th preliminary location $\boldsymbol{\mu}_n^{(t+1)*}$, for $n = 1, \cdots, N$, is resampled from the set $\{\mathbf{x}_{n,k}^{(t)}\}_{k=1}^K$ using the following normalized weight as the multinomial probability mass function:

$$\bar{w}_{n,k}^{(t)} = \frac{w_{n,k}^{(t)}}{\sum_{i=1}^K w_{n,i}^{(t)}}. \quad (15)$$

After resampling, a new set of preliminary locations for the next iteration, $P = \{\boldsymbol{\mu}_1^{(t+1)*}, \boldsymbol{\mu}_2^{(t+1)*}, \cdots, \boldsymbol{\mu}_N^{(t+1)*}\}$, is

obtained. There are various resampling methods in the literature [8], [34]. Here we use the simple multinomial LR scheme where it generates N preliminary locations that are different, thereby preserving the diversity in exploring the target space (see Fig. 3 for a visual illustration of how LR scheme operates).

ii) Preliminary locations generation based on proposal locations

We generate the second set of N preliminary locations from the currently available proposal locations, as shown in the top-left block of the adaptation module in Fig. 2. This process begins by running N independent Hamiltonian Monte Carlo (HMC) methods in parallel. Each HMC method, represented as an HMC block in Fig. 4, generates a chain with $\boldsymbol{\mu}_n^{(1)}$ as the initial starting point. These N parallel HMC blocks then produce the new set of N preliminary locations, $Q = \{\boldsymbol{\mu}_1^{(t+1)*}, \boldsymbol{\mu}_2^{(t+1)*}, \cdots, \boldsymbol{\mu}_N^{(t+1)*}\}$, by applying one iteration of N parallel chains. We can consider $\widetilde{\pi}^{(t+1)*}(\boldsymbol{\mu})$ as a random measure that approximates the target distribution, i.e.,

$$\widetilde{\pi}^{(t+1)*}(\boldsymbol{\mu}) = \sum_{n=1}^N \bar{w}_{\boldsymbol{\mu}_n^{(t+1)*}} \delta(\boldsymbol{\mu} - \boldsymbol{\mu}_n^{(t+1)*}), \quad (16)$$

where $\bar{w}_{\boldsymbol{\mu}_n^{(t+1)*}}$ is the normalized DM weight of each $\boldsymbol{\mu}_n^{(t+1)*}$. The theoretical justification is that, following the burn-in period, the N parallel HMC chains have reached convergence with the target distribution, so $\boldsymbol{\mu}_n^{(t+1)*} \sim \pi$. For more details we refer the reader to our previous HAIS method [25].

Step 3(b): Final locations generation through cooperation

At this step, the goal is to determine the final proposal locations for the next iteration. Let us denote the population of all preliminary locations from Step 3(a) as

$$C = \{P \cup Q\}.$$

The cooperation procedure takes as input the $|C| = 2N$ preliminary locations (where $|.|$ denotes the cardinality operator) from the previous step and outputs N final locations for the next iteration $(t + 1)$. This process is depicted in the bottom block of the adaptation module in Fig. 2. We propose two different methods for this process: cooperation by *weighted mixture model* and by *resampling*.

a) Cooperation by weighted mixture model: Here the set of $|C|$ preliminary locations is used to construct a weighted mixture model. This model serves as a proper proposal PDF to generate new samples, which are the final proposal locations for the next iteration. Let us define a weighted mixture of PDFs as

$$\psi(\mu) = \sum_{i=1}^{|C|} \bar{w}_i \varphi_i(\mu | \mu_i^{(t+1)*}), \quad (17)$$

where the i-th component φ_i is a parametric kernel function with two parameters, e.g. mean and covariance in the Gaussian case. Additionally, μ_i is the i-th preliminary location from Step 1 serving as the location parameter of the i-th component. Here w_i is the DM weight of i-th component φ_i, which is

Fig. 2: The flowchart of the proposed HPMC method illustrates the three main operations at iteration t, which lead to the estimated values. The process consists of three primary modules—sampling, weighting, and adaptation—along with initialization and output modules.
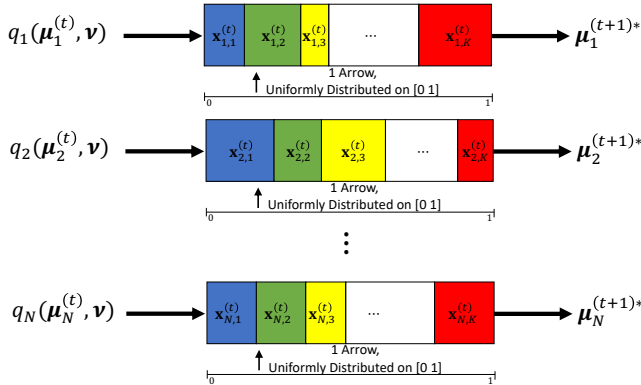

Fig. 3: LR Resampling Scheme: A set of K weighted samples, simulated from a proposal, is resampled to generate a new preliminary location for the next step. Each rectangle represents a sample, with its length corresponding to its normalized weight. A vertical arrow indicates a single random uniform sample drawn in each row.
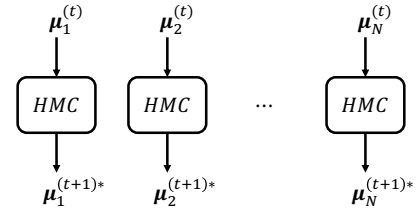

Fig. 4: Preliminary locations generation based on current locations: N independent HMC blocks operate in parallel to generate the new set of N preliminary locations.

$$w_i = \frac{\pi(\boldsymbol{\mu}_i^{(t+1)*})}{\sum_{j=1}^{|C|} q_j(\boldsymbol{\mu}_j^{(t+1)*}; \boldsymbol{\mu}_j^{(t)}, \nu_j)}, \quad i = 1, \cdots, |C|, \quad (18)$$

and then normalized by

$$\bar{w}_i = \frac{w_i}{\sum_{j=1}^{|C|} w_j}, \quad i = 1, \cdots, |C|.$$

$\psi(\mu)$ is a suitable proposal density for generating new samples and is updated at each AIS iteration. The theoretical justification is that, after the preliminary locations generation step, all the $\mu_i^{(t+1)*}$ have converged to the target PDF, i.e., $\mu_i^{(t+1)*} \sim \pi$, $i = 1, \cdots, |C|$. As the result, $\psi(\mu)$ can be

viewed as a weighted nonparametric kernel estimation of the target π. The iterative algorithm starts by generating one sample from $\psi(\mu)$ as a new candidate location, i.e., $\{\mu'_j\} \sim \psi(\mu)$. In the next step, μ'_j is accepted with a probability value α_j then is considered as the final proposal location for the next iteration of the AIS mechanism. Alg. 2 provides a detailed description of the proposed cooperation step.

b) Cooperation by resampling: We can define $\widetilde{\pi}(\boldsymbol{\mu})$ as a random measure based on $|C|$ preliminary locations from the previous step as,

$$\widetilde{\pi}(\boldsymbol{\mu}) = \sum_{i=1}^{|C|} \bar{w}_i \delta(\boldsymbol{\mu} - \boldsymbol{\mu}_i^{(t+1)*}), \quad (19)$$

where $\bar{w}_i$ is the normalized DM weight of i-th preliminary location, obtained from Eq. (18). The final proposal locations for the next iteration are then obtained by sampling from the random measure $\widetilde{\pi}(\boldsymbol{\mu})$ via global resampling (GR) [13], i.e., $\boldsymbol{\mu}_n^{(t+1)} \sim \widetilde{\pi}(\boldsymbol{\mu})$, $n = 1, \cdots, N$. In GR, all the preliminary

---

Algorithm 2: Cooperation by weighted mixture model

Input: a set of $|C|$ preliminary proposal locations from the previous step, $\{\mu_i^{(t+1)*}\}_{i=1}^{|C|} \in C$

Build the weighted mixture model $\psi(\mu)$ based on Eq. (17)

for $j = 1, ..., N$ **do**

    Sample $\mu'_j \sim \psi(\mu)$

    Set $\mu_j^{(t+1)} = \mu'_j$ with probability

$$\alpha_j = \min\left(\frac{\pi(\mu'_j)\psi(\mu_j^{(t+1)*})}{\pi(\mu_j^{(t+1)*})\psi(\mu'_j)}\right)$$

    Otherwise set

$$\mu_j^{(t+1)} = \mu_j^{(t+1)*}$$

end

Output: final set of proposal locations, $\mu_j^{(t+1)}$, $i = 1, \cdots, N$ for the next iteration

---

locations from the two sets, P and Q, are resampled together. To better understand the idea of GR scheme see Fig. 5.
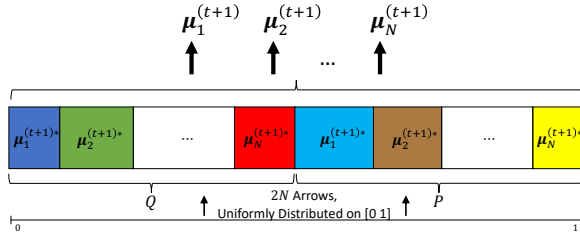

Fig. 5: GR Resampling Scheme: A set of $2N$ weighted preliminary locations is resampled to generate the final location for the next iteration. Each rectangle represents a sample, with the length of the rectangle corresponding to its normalized weight. Vertical arrows indicate NN random uniform samples.

After resampling, a new set of final locations for the next iteration, $\{\mu_1^{(t+1)}, \mu_2^{(t+1)}, \cdots, \mu_N^{(t+1)}\}$, is obtained, and a modified and unweighted random measure is produced as

$$\widetilde{\pi}^{(t+1)}(\mu) = \frac{1}{N}\sum_{n=1}^{N}\delta(\mu - \mu_n^{(t+1)}). \qquad (20)$$

Fig. 6 displays the proposed two-step adaptation mechanism using a univariate bi-modal target as a toy example. The new weighted proposal locations (depicted with red points as the proposal locations and red vertical lines as the corresponding weights) are obtained by performing the preliminary location generation step on the proposal locations from iteration (t). However, in this case, the probability assigned to each mode in the resulting proposal PDF is not balanced, and thus it may not accurately reflect the target distribution. Subsequently, the probability of each mode is adjusted through the cooperation step, where the final unweighted proposal locations for the next iteration $(t+1)$ are generated.

D. Discussion

In HPMC, a two-step hybrid adaptation process is presented: a preliminary location generation step and a cooperation step. Here, we discuss the key components of the HPMC algorithm:
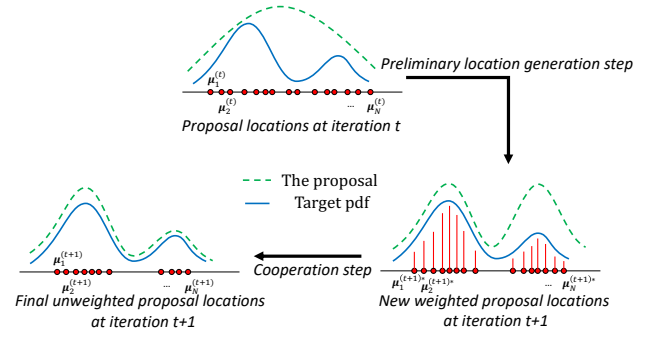

Fig. 6: The procedure of the HPMC adaptation step of the mixture of proposals (in green dashed line) in a typical one-dimensional bimodal target PDF (in blue solid lines). The red points are the locations of the proposals, and the red vertical lines are proportional to the importance weights.

a) Preliminary location generation (local-global exploration trade-off): In HPMC, we propose two different operations to generate preliminary locations:

In the first operation, N new preliminary locations are generated via down-sampling from NK DM-weighted samples using a local resampling (LR) scheme. The LR approach ensures that only one preliminary location generated from each proposal location is retained. In the DM weighting applied here, the entire set of proposals in the denominator is evaluated. This interaction between proposals allows higher weights to be assigned to samples in under-populated or unexplored regions, increasing the likelihood of selecting preliminary locations from those areas. Consequently, the LR resampling with DM weights enhances the local exploration capability of the HMPC estimator, particularly in high-dimensional multi-modal target distributions. Increasing the number of samples per proposal, K, further improves the likelihood of discovering local relevant features of the target.

In the second operation, N independent HMC blocks run in parallel to generate N preliminary locations for the next step. Each HMC block, with its exploratory behavior, especially in high-dimensional scenarios, acts as a *free explorer* navigating the target state space. Increasing the number of these explorers enhances the global exploration capability of the estimator. Consequently, for a fixed number of target evaluations, there is a trade-off between promoting local exploration (by increasing K) and enhancing global exploration (by increasing N). By leveraging both weighted samples and location parameters, HPMC reduces the number of iterations required to achieve the target estimate, leading to faster convergence.

b) Cooperation: In this step, information is exchanged among the preliminary locations to redirect the lost explorers toward the optimal path based on the target PDF. Cooperation is performed using two different methods, both applying the DM weighting approach to assign weights to the preliminary locations from the previous step. The DM weighting enforces a repulsive behavior among the locations by using the complete mixture of all proposals in the denominator, as discussed earlier. In other words, locations that are less covered by the remaining $|C| - 1$ locations receive higher weights, enhancing the exploratory behavior of the cooperation step.

In the resampling method, the GR scheme with global DM weights is used to terminate lost explorers in irrelevant regions of the target, while replicating more promising ones—those with higher global weights.

IV. NUMERICAL EXAMPLES

In this section, we evaluate the efficiency of the proposed method through three experiments and compare its performance with existing methods in the literature. The first experiment considers a two-dimensional multi-modal target as a toy example. The second experiment evaluates a high-dimensional bimodal target probability density function (PDF). Finally, the third experiment involves a challenging benchmark problem for Monte Carlo (MC) algorithms: sampling from a banana-shaped distribution at different dimensions. We also compare the computational complexity of the proposed method with other exiting algorithms in terms of K (number of samples generated from each proposal PDF), N (number of proposals PDFs), and T (number of iterations).

In all experiments, the number of iterations, T, for each method is set such that they have the same total number of target evaluations, $E = 2 \times 10^5$. We compute the mean squared error (MSE) in the estimation of $\mathbb{E}[\mathbf{x}]$ and the normalizing constant Z of the target. The results are averaged over 200 Monte Carlo simulations.

We compare the proposed hybrid population Monte Carlo (HPMC) techniques with

- standard PMC [6];
- DM-PMC, GR-PMC and LR-PMC methods [13];
- AMIS [16] with one single proposal PDF ($N = 1$);
- PI-MAIS [23] where for the upper layer, we also consider Gaussian PDFs $\varphi_n(\mathbf{x}|\boldsymbol{\mu}_n, \Lambda_n)$ where covariance matrices $\Lambda_n = \lambda^2 I_2$ with $\lambda \in \{5, 10\}$;
- HAIS [25] with $\epsilon \in \{5, 10\}$ and $L = 50$.

A. Toy example

We consider a two-dimensional example to examine how different methods behave along iterations. The target is a mixture of five bivariate Gaussian PDFs

$$\pi(\mathbf{x}) = \frac{1}{5} \sum_{i=1}^{5} \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i), \ \mathbf{x} \in \mathbb{R}^2, \tag{21}$$

where the vector of means are $\boldsymbol{\mu}_1 = [-10, -10]^T$, $\boldsymbol{\mu}_2 = [0, 16]^T$, $\boldsymbol{\mu}_3 = [13, 8]^T$, $\boldsymbol{\mu}_4 = [-9, 7]^T$, $\boldsymbol{\mu}_5 = [14, -14]^T$, and the covariance matrices are $\boldsymbol{\Sigma}_1 = [2, 0.6; 0.6, 1]^T$, $\boldsymbol{\Sigma}_2 = [2, -0.4; -0.4, 2]^T$, $\boldsymbol{\Sigma}_3 = [2, 0.8; 0.8, 2]^T$, $\boldsymbol{\Sigma}_4 = [3, 0; 0, 0.5]^T$, and $\boldsymbol{\Sigma}_5 = [2, -0.1; -0.1, 2]^T$. This is a challenging example due to difficulty in discovering all five different modes, particularly if one selects a bad initialization for proposals. We use isotropic Gaussian proposals with locations within the square $[-4, 4] \times [-4, 4]$, i.e., $\{\boldsymbol{\mu}_n^{(1)}\}_{n=1}^{N} \sim U([-4, 4] \times [-4, 4])$, where all of the modes in the target PDF are out of the initialization area. We also use the same isotropic covariance matrices for all the proposals, $\{\boldsymbol{\Sigma}_n\}_{n=1}^{N} = \sigma^2 I_2$ with $\sigma = 5$. We set $N = 100$ proposals, and $K = 5$ sample per proposal. We run GR-PMC and LR-PMC methods [13],

PI-AMIS [23], and HPMC with the same setting for a fair comparison. Fig. 7 shows the graphical evolution through the first three iterations of the proposal locations (black circles) and generated samples (red crosses). In HPMC, the proposal locations rapidly discover all five modes in the first iteration, and an approximately uniform distribution of the proposal locations around all the target modes is achieved in the next iterations. So an accurate target estimation is obtained after a few iterations. PI-AMIS almost finds all five modes after three iterations, showing slow convergence, as the proposals are located between the initial positions and the centers of the target modes. One limitation of PI-AMIS is the well-known random-walk behavior of the MH that makes the convergence of the Markov chain inefficient. Both LR-PMC and GR-PMC are failed to estimate the target accurately. The local resampling technique in LR-PMC maintains the diversity of the proposal population, leading to the discovery of all target modes. However, it lacks global approximation because the proposal locations are unevenly distributed. In GR-PMC, after three iterations, the proposals are almost distributed around four modes while one mode is completely missed.

B. Bimodal target distribution in high dimension

In this section, we consider sampling from a high-dimensional bi-modal target PDF as a challenging problem. The target is a mixture of two Gaussians, i.e., $\pi(\mathbf{x}) = \sum_{i=1}^{2} \frac{1}{2} \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$. Here $\mathbf{x} \in \mathbb{R}^{20}$, $\boldsymbol{\mu}_i = [\mu_{i,1}, \mu_{i,2}, ..., \mu_{i,20}]^T$, and $\boldsymbol{\Sigma}_i = c\mathbf{I}_{20}$, for $i \in \{1, 2\}$ where $\mathbf{I}_{20}$ is an identity matrix of dimension 20. We choose $\mu_{1,d} = 8$ and $\mu_{2,d} = -8$ for all dimensions and also we set $c = 5$. Multi-modal settings are challenging, and in this example the two modes are distant, which over-complicates the exploration process at high dimension. Moreover, the initial starting points play a crucial role in exploring the target. Here, we consider a cold-start scenario to evaluate robustness and adaptation capabilities. For simplicity, we assume the proposal densities are Gaussian PDFs with uniformly selected initial means, i.e., $\boldsymbol{\mu}_n \sim U([-4 \times 4]^{d_x})$ for $n = 1, ..., N$, where none of the modes of the target fall within this area. We use the same isotropic covariance for all of proposals, $\boldsymbol{\Sigma} = \sigma^2 \mathbf{I}_{20}$ with $\sigma \in \{1, 2, 5, 10, 20\}$.

We compute MSE in the estimation of the mean and the normalizing constant of the target, which can be find analytically as $\mathbb{E}[\mathbf{x}] = 0$ and $Z = 1$. The MSE in the estimation of $\mathbb{E}[\mathbf{x}]$ is averaged over all 20 components. All algorithms are simulated with a fixed number of total generated samples in T iterations, i.e., $NK = 500$. We test for two scenarios, once we choose $N = 100$ and $K = 5$ and in the second scenario we select $N = 250$ and $K = 2$. The simulation results from each method are summarized in Table I in terms of MSE in the estimation of $\mathbb{E}[\mathbf{x}]$ and Z for different configurations of N, K and various range of σ values.

First, note that most algorithms result in very large MSE values. These situations often occur when one or both modes fail to be discovered. When both modes are missed, the estimation of normalizing constant of the target is $\widehat{Z} \approx 0$ which corresponds to a MSE≈ 1, as it happens in several settings.
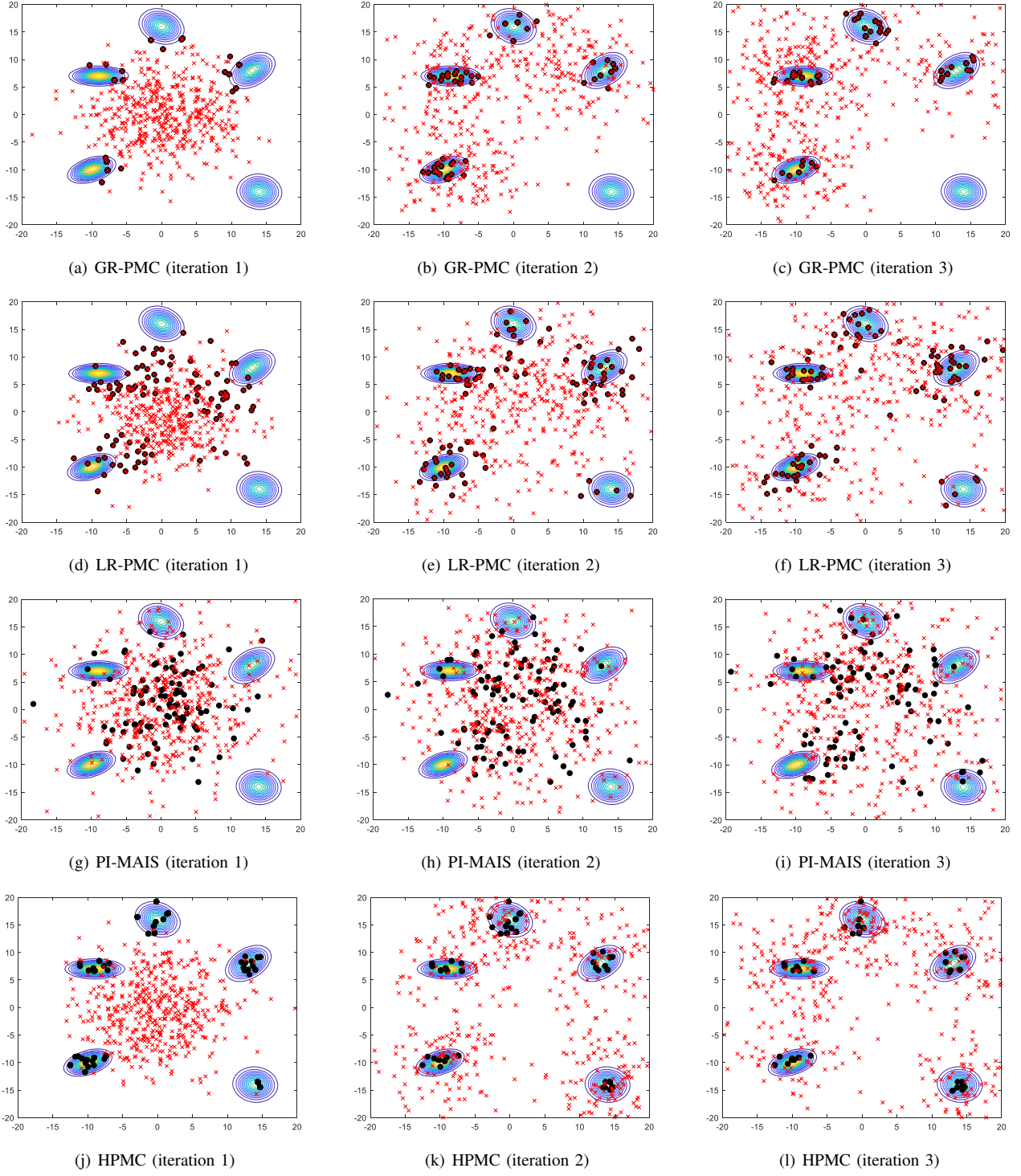
Fig. 7: Toy example: Evolution of proposal locations (black circles) and samples (red crosses) along three iterations for GR-PMC, PI-MAIS, and HPMC using $N = 100$ and $K = 5$. The multi-modal target density with five modes is shown in contours in 2D plots.

Second, we observe that the two proposed schemes outperform the other methods. According to this table, HPMC algorithm with cooperation by resampling generally achieves the smallest MSE values for the mean and normalizing constant estimation, ($\text{MSE}_{\mathbb{E}[\mathbf{x}]} = 7.7889$ and $\text{MSE}_Z = 0.0009$), where $N = 250$, $K = 2$, and $\epsilon = 5$. It is evident that increasing N has a greater impact on MSE reduction than increasing K. The

intuition behind this is that increasing the number of proposals, with each proposal acting as an independent explorer of the target space, enhances the global exploration capability of the sampling method. We note that the optimal value of σ, which yields the smallest MSE, depends on the scale parameter of the target (here it is $c = 5$).

TABLE I: MSE in the approximation of the mean and normalizing constant of multi-modal target distribution. The best results for each value of σ are boldfaced

Method	$\sigma = 1$		$\sigma = 2$		$\sigma = 5$		$\sigma = 10$		$\sigma = 20$	
	$\mathbb{E}[\mathbf{x}]$	Z	$\mathbb{E}[\mathbf{x}]$	Z	$\mathbb{E}[\mathbf{x}]$	Z	$\mathbb{E}[\mathbf{x}]$	Z	$\mathbb{E}[\mathbf{x}]$	Z
Standard PMC $N = 100, K = 1$	70.1248	1	71.0234	1	69.7048	1	78.3254	1	152.3154	1
DM-PMC $N = 100, K = 1$	66.2487	1	67.0015	0.9999	66.3021	1	69.3215	1	149.0478	1
LR-PMC										
$N = 100, K = 5$	21.5680	0.9974	26.7301	0.0998	65.2457	0.9997	65.0219	1	137.2168	1
$N = 250, K = 2$	20.1789	0.9545	24.8894	0.0754	46.2145	0.9632	53.0985	1	131.0245	1
GR-PMC										
$N = 100, K = 5$	64.9203	0.9418	64.1398	0.3126	65.7060	0.3412	83.9430	1	143.4574	1
$N = 250, K = 2$	61.0387	0.9105	62.0985	0.2985	56.2015	0.3101	71.0215	1	140.2015	1
AMIS $N = 1, K = 500$	80.2054	1	79.0548	0.9998	82.0195	0.9996	102.0215	1	159.0598	1
PI-MAIS										
$N = 100, K = 5$										
$\lambda = 5$	46.1009	1	38.9915	0.4531	36.6260	0.5421	65.5964	0.9389	135.5524	1
$\lambda = 10$	60.5114	1	60.3086	1	53.2838	0.9878	61.7771	0.9946	130.5375	1
$N = 250, K = 2$										
$\lambda = 5$	41.1068	0.9568	37.0658	0.3057	31.0287	0.3924	61.0548	0.9702	131.2050	1
$\lambda = 10$	43.5145	0.9328	45.0916	0.8576	35.0367	0.5205	62.0348	0.9905	132.0957	1
HAIS										
$N = 100, K = 5$										
$\epsilon = 5$	41.0925	0.8649	17.5741	0.0201	13.1954	0.0031	58.0607	0.8573	109.0967	0.9899
$\epsilon = 10$	42.7641	0.8828	17.3210	0.0162	12.8675	0.0016	61.5772	0.9823	117.7326	1
$N = 250, K = 2$										
$\epsilon = 5$	32.0568	0.7565	15.0248	0.0169	9.0249	0.0013	41.0254	**0.7535**	**99.0320**	0.9855
$\epsilon = 10$	34.9875	0.7575	17.0254	0.0113	11.0587	0.0083	48.0548	0.7814	104.0032	0.9917
HPMC (cooperation by mixture model)										
$N = 100, K = 5$										
$\epsilon = 5$	39.0215	0.8816	17.0248	0.0209	16.2054	0.0990	55.2213	0.8641	113.0558	0.9888
$\epsilon = 10$	40.0215	0.8791	19.8332	0.0193	15.9683	0.0117	60.3506	0.9401	117.9220	0.9990
$N = 250, K = 2$										
$\epsilon = 5$	33.3467	0.7639	15.2418	0.0161	10.5921	0.0088	49.6691	0.8857	110.9152	0.9989
$\epsilon = 10$	34.0122	0.7501	16.9735	0.0126	11.9617	0.0107	50.1877	0.7819	112.2142	0.9901
HPMC (cooperation by resampling)										
$N = 100, K = 5$										
$\epsilon = 5$	21.7384	0.7508	15.2681	0.0217	8.4948	0.0018	45.7573	0.8820	110.3492	0.9809
$\epsilon = 10$	22.7792	0.7444	17.5407	0.0170	9.4434	0.0015	44.3853	0.8925	112.14.15	0.9878
$N = 250, K = 2$										
$\epsilon = 5$	**19.3494**	**0.6211**	**14.5010**	0.0127	**7.7889**	**0.0009**	**40.9551**	0.7591	100.1874	**0.9790**
$\epsilon = 10$	21.7117	0.6496	15.6816	**0.0111**	9.8156	0.0011	43.1310	0.7747	101.8411	0.9784

C. High-dimensional banana-shaped target distribution

We now consider a benchmark multidimensional banana-shaped target distribution [35], which is a challenging example because of its nonlinear nature, specially in high dimension. The target PDF is given by

$$\bar{\pi}(x_1, ..., x_{d_x})$$
$$\propto \exp\left(-\frac{x_1^2}{2\sigma^2} - \frac{\left(x_2 + b(x_1^2 - \sigma^2)\right)^2}{2\sigma^2} - \sum_{i=3}^{d_x} \frac{x_i^2}{2\sigma^2}\right). \quad (22)$$

We set $b = 3$ and $\sigma = 1$, then the true value for $\mathbb{E}_{\bar{\pi}}[\mathbf{x}] = 0$. In this experiment, we use Gaussian proposals with initial locations similar to those in the previous experiment. An isotropic covariance matrix is considered for all the proposal PDFs, $\mathbf{\Sigma} = \sigma^2 \mathbf{I}_{d_x}$ with $\sigma^2 = 1$. All algorithms are simulated with a fixed number of samples per proposals, i.e., $K = 5$. However, we test two different values for the number of proposals, $N = 100$ and $N = 200$ to asses the impact of increasing N on the performance of the estimator. As in the previous experiment, to ensure a fair comparison, we select

T such that all algorithms have the same total number of target evaluations, $E = 2 \times 10^5$. We compute the MSE in the estimation of $\mathbb{E}[\mathbf{x}]$ and the results are averaged over 200 Monte Carlo simulations. To compare the performance of the proposed method with other approaches as the dimension of the state space increases, we vary the dimension of the state space by testing different values of d_x (with $2 \leq d_x \leq 50$). Fig. 8 shows the MSE in the estimation of $E[\mathbf{x}]$ as a function of the dimension d_x of the state-space, regarding the same techniques as in the previous bi-modal example. The performance of all methods degrades as the dimension increases. The result indicates that the novel HPMC with resampling scheme outperforms all the other methods under fair computational complexity comparison. In Fig. 8-b we doubled the number of proposal PDFs (to $N = 200$) to see how the MSE values change across different dimensions. This adjustment leads to a significant decrease in MSE across all dimensions for all methods, with the proposed HPMC with the resampling method showing particularly notable improvements.
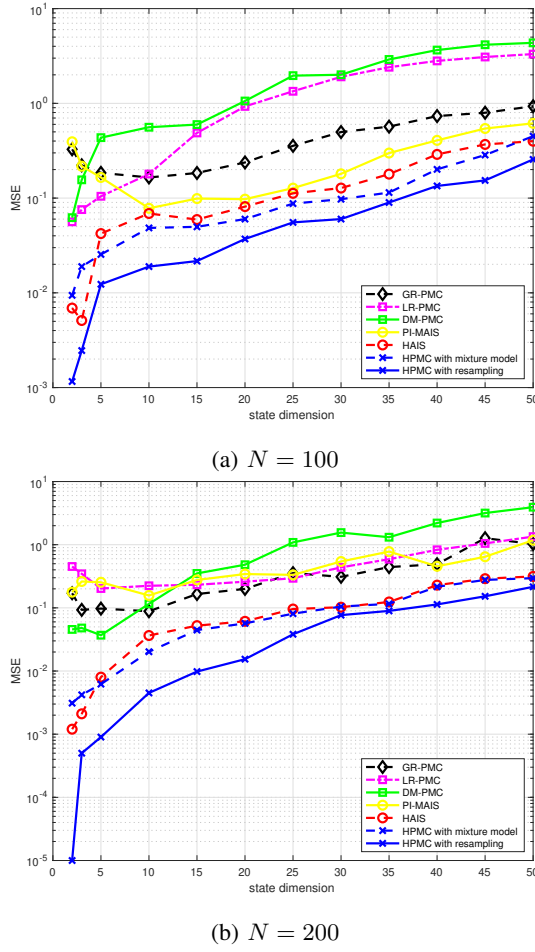
(a) $N = 100$

(b) $N = 200$

Fig. 8: MSE of $\mathbb{E}[\mathbf{x}]$, using two different values for N as the number of proposals with $\sigma = 1$, as the dimension of the target PDF, d_x, increases.

D. Computational complexity comparison

In this section, we provide a comparison of the computational complexity of different methods. In the MC domain, computational complexity is described as the total number of evaluations of proposal and target PDFs, as well as the computational cost per each generated sample. However, the number of target evaluations is of greater importance because, in most MC methods, the proposal PDF is typically chosen to be a Gaussian distribution, which is simpler to evaluate compared to complex target PDFs. Table II compares the computational complexity of different methods in terms of three parameters: K (number of samples generated from each proposal PDF), N (number of proposals PDFs), and T (number of iterations). In the DM weighting technique, N proposal evaluations are required for each sample. As shown in Table II, all methods that use the DM weighting technique have KN^2T as the number of proposal evaluations for the total NKT generated samples. $3NT$ and $2NT$ are the number of target evaluations that occur in the cooperation by mixture model and resampling, respectively through iterations. As shown in the table, the number of target evaluations for the HPMC by resampling is lower than that for HPMC by mixture model. It can be noticed that the computational complexity of the

proposed HPMC (mixture model) method is higher than that of all other algorithms. The other HPMC (with resampling), however, requires fewer computations, and its complexity is comparable to that of PI-MAIS and HAIS, two recent works in the literature.

V. CONCLUSION

In this paper, we have proposed the hybrid population Monte Carlo (HPMC) algorithm, a hybrid adaptive importance sampler with strong performance for high-dimensional and multi-modal problems. HPMC belongs to the population Monte Carlo (PMC) algorithms and incorporates a two-step hybrid adaptation mechanism. In the first step, preliminary locations are generated based on weighted samples and location parameter of the proposals. This hybrid approach enhances diversity, promotes both local and global exploration, and accelerates convergence. Notably, the novel cooperation step exchanges information among preliminary locations, further improving global exploration. The method is theoretically justified through its hybrid adaptation procedure and the cooperation techniques. Simulation results have shown significant improvements compared with other state-of-the-art methods. Future lines include the development of alternative adaptation mechanisms for the location parameters and the extension of the approach to adapt the covariance of the proposals.

REFERENCES

[1] C. Robert and G. Casella, "Monte carlo statistical methods. 2004."
[2] J. S. Liu and J. S. Liu, *Monte Carlo strategies in scientific computing*, vol. 10. Springer, 2001.
[3] A. B. Owen, "Monte carlo theory, methods and examples," 2013.
[4] V. Elvira, L. Martino, D. Luengo, M. F. Bugallo, *et al.*, "Generalized multiple importance sampling," *Statistical Science*, vol. 34, no. 1, pp. 129–155, 2019.
[5] M.-S. Oh and J. O. Berger, "Adaptive importance sampling in monte carlo integration," *Journal of Statistical Computation and Simulation*, vol. 41, no. 3-4, pp. 143–168, 1992.
[6] O. Cappé, A. Guillin, J.-M. Marin, and C. P. Robert, "Population monte carlo," *Journal of Computational and Graphical Statistics*, vol. 13, no. 4, pp. 907–929, 2004.
[7] R. Douc and O. Cappé, "Comparison of resampling schemes for particle filtering," in *ISPA 2005. Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis, 2005.*, pp. 64–69, Ieee, 2005.
[8] T. Li, M. Bolic, and P. M. Djuric, "Resampling methods for particle filtering: classification, implementation, and strategies," *IEEE Signal processing magazine*, vol. 32, no. 3, pp. 70–86, 2015.
[9] M. F. Bugallo, V. Elvira, L. Martino, D. Luengo, J. Miguez, and P. M. Djuric, "Adaptive importance sampling: the past, the present, and the future," *IEEE Signal Processing Magazine*, vol. 34, no. 4, pp. 60–79, 2017.
[10] V. Elvira and L. Martino, "Advances in importance sampling," *arXiv preprint arXiv:2102.05407*, 2021.
[11] E. Koblents and J. Míguez, "A population monte carlo scheme with transformed weights and its application to stochastic kinetic models," *Statistics and Computing*, vol. 25, no. 2, pp. 407–425, 2015.
[12] O. Cappé, R. Douc, A. Guillin, J.-M. Marin, and C. P. Robert, "Adaptive importance sampling in general mixture classes," *Statistics and Computing*, vol. 18, no. 4, pp. 447–459, 2008.
[13] V. Elvira, L. Martino, D. Luengo, and M. F. Bugallo, "Improving population monte carlo: Alternative weighting and resampling schemes," *Signal Processing*, vol. 131, pp. 77–91, 2017.
[14] V. Elvira and E. Chouzenoux, "Langevin-based strategy for efficient proposal adaptation in population monte carlo," in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5077–5081, IEEE, 2019.

TABLE II: Computational complexity comparison of different methods.

Algorithm	Number of target evaluations	Number of proposal evaluations	Number of target evaluations per sample	Number of proposal evaluations per sample
Standard PMC	NT	NT	1	1
DM-PMC	KNT	KN^2T	1	N
N-PMC	NT	NT	1	1
M-PMC	KT	KNT	1	N
LR-PMC	KNT	KN^2T	1	N
GR-PMC	KNT	KN^2T	1	N
AMIS	KT	KT^2	1	T
PI-MAIS	$KNT + NT$	KN^2T	$1 + 1/K$	N
APIS	KNT	KN^2T	1	N
GAPIS	KNT	KN^2T	1	N
HAIS	$KNT + NT$	KN^2T	$1 + 1/K$	N
HPMC (mixture model)	$KNT + 3NT$	KN^2T	$1 + 2/K + 1/KT$	N
HPMC (resampling)	$KNT + 2NT$	KN^2T	$1 + 2/K$	N

[15] V. Elvira and E. Chouzenoux, "Optimized population monte carlo," *IEEE Transactions on Signal Processing*, vol. 70, pp. 2489–2501, 2022.

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

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

[18] T. L. T. Nguyen, F. Septier, G. W. Peters, and Y. Delignon, "Efficient sequential monte-carlo samplers for bayesian inference," *IEEE Transactions on Signal Processing*, vol. 64, no. 5, pp. 1305–1319, 2015.

[19] M. Karamanis and U. Seljak, "Persistent sampling: Unleashing the potential of sequential monte carlo," *arXiv preprint arXiv:2407.20722*, 2024.

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

[21] I. Schuster, "Gradient importance sampling," tech. rep., 2015. https://arxiv.org/abs/1507.05781.

[22] M. Fasiolo, F. E. de Melo, and S. Maskell, "Langevin incremental mixture importance sampling," *Stat. Comput.*, vol. 28, no. 3, pp. 549–561, 2018.

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

[24] I. Klebanov and I. Schuster, "Markov chain importance sampling-a highly efficient estimator for mcmc," *Journal of Computational and Graphical Statistics*, 2020.

[25] A. Mousavi, R. Monsefi, and V. Elvira, "Hamiltonian adaptive importance sampling," *IEEE Signal Processing Letters*, vol. 28, pp. 713–717, 2021.

[26] S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth, "Hybrid monte carlo," *Physics letters B*, vol. 195, no. 2, pp. 216–222, 1987.

[27] R. M. Neal *et al.*, "Mcmc using hamiltonian dynamics," *Handbook of markov chain monte carlo*, vol. 2, no. 11, p. 2, 2011.

[28] E. Veach and L. J. Guibas, "Optimally combining sampling techniques for monte carlo rendering," in *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pp. 419–428, 1995.

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

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

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

[32] V. Elvira, L. Martino, D. Luengo, and M. F. Bugallo, "Population monte carlo schemes with reduced path degeneracy," in *2017 IEEE 7th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pp. 1–5, IEEE, 2017.

[33] L. Martino, V. Elvira, and D. Luengo, "Anti-tempered layered adaptive importance sampling," in *2017 22nd International Conference on Digital Signal Processing (DSP)*, pp. 1–5, IEEE, 2017.

[34] C. Kuptametee and N. Aunsri, "A review of resampling techniques in particle filtering framework," *Measurement*, vol. 193, p. 110836, 2022.

[35] H. Haario, E. Saksman, J. Tamminen, *et al.*, "An adaptive metropolis algorithm," *Bernoulli*, vol. 7, no. 2, pp. 223–242, 2001.