

# Gradient-based Adaptive Importance Samplers

Víctor Elvira<sup>a,\*</sup>, Émilie Chouzenoux<sup>b</sup>, Ömer Deniz Akyildiz<sup>c</sup>, Luca Martino<sup>d</sup>

<sup>a</sup>*School of Mathematics, University of Edinburgh, UK.*

<sup>b</sup>*CVN, INRIA Saclay, CentraleSupélec, France.*

<sup>c</sup>*Dept. of Mathematics, Imperial College London, UK.*

<sup>d</sup>*Universidad Rey Juan Carlos, Spain.*

---

## Abstract

Importance sampling (IS) is a powerful Monte Carlo methodology for the approximation of intractable integrals, very often involving a target probability distribution. The performance of IS heavily depends on the appropriate selection of the proposal distributions where the samples are simulated from. In this paper, we propose an adaptive importance sampler, called GRAMIS, that iteratively improves the set of proposals. The algorithm exploits geometric information of the target to adapt the location and scale parameters of those proposals. Moreover, in order to allow for a cooperative adaptation, a repulsion term is introduced that favors a coordinated exploration of the state space. This translates into a more diverse exploration and a better approximation of the target via the mixture of proposals. Moreover, we provide a theoretical justification of the repulsion term. We show the good performance of GRAMIS in two problems where the target has a challenging shape and cannot be easily approximated by a standard uni-modal proposal.

*Keywords:* Adaptive importance sampling, Monte Carlo, Bayesian inference, Langevin adaptation, Poisson field, Gaussian mixture

---

## 1. Introduction

The approximation of intractable integrals is a common statistical task in many applications of science and engineering. A relevant example is the case of Bayesian

---

\*Corresponding author

*Email address:* victor.elvira@ed.ac.uk (Víctor Elvira)

inference arising for instance in statistical machine learning. A posterior distribution of unknown parameters is constructed by combining data (through a likelihood model) and previous information (through the prior distribution). Unfortunately, the posterior is often unavailable, typically due to the intractability of the marginal likelihood.

Monte Carlo methods have proved to be effective in those relevant problems, producing a set of random samples that can be used to approximate a target probability distribution and integrals related to it [1, 2, 3]. Importance sampling (IS) is one of the main Monte Carlo families, with solid theoretical guarantees [3, 4]. The vanilla version of IS simulates samples from the so-called proposal distribution. Then, each sample receives an associated weight which is computed by taking into account the mismatch between this proposal probability density function (pdf) and the target pdf. While the IS mechanism is valid under very mild assumptions [3, 4], the efficiency of the estimators is driven by the choice of the proposal distribution. Unfortunately this choice is a difficult task, even more in contexts where one has access to the evaluation of an unnormalized version of the posterior distribution, as it is the case in Bayesian inference. The two main methodological directions to overcome the limitations in IS are the usage of several proposals, which is known as multiple IS (MIS) [5], and the iterative adaptation of those proposals, known as adaptive importance sampling (AIS) [6].

There exist a plethora of AIS algorithms, and we refer the interested reader to [6]. There are also strong theoretical guarantees for some subclasses of AIS algorithms, see, e.g., [7, 8, 9]. These AIS methods can be arguably divided in three main categories. The first category is based on sequential moment matching and includes algorithms that implement Rao-Blackwellization in the temporal estimators ([10, 11]), extend to the MIS setting [12], or are based on a sequence of transformations that can be interpreted as a change of proposal [13]. A second AIS family comprises the population Monte Carlo (PMC) methods which use resampling mechanisms to adapt the proposals [14, 15]. The PMC framework was first introduced in [16] and then extended by the incorporation of stochastic expectation-maximization mechanisms [17], clipping of the importance weights [18, 19], improving the weighting and resampling mechanisms [20, 21], targeting the estimation of rare-event probabilities [22], or introducing

optimization schemes [23].

Finally, a third category contains AIS methods with a hierarchical or layered structure. Examples of these algorithms are those that adapt the location parameters of the proposals using a Markov chain Monte Carlo (MCMC) mechanism [24, 25, 26, 27]. In this category, we also include methods that exploit geometric information about the target for the adaptation of the location parameters, yielding to optimization-based adaptation schemes. In the layered mechanism, the past samples do not affect the proposal adaptation which is rather driven by the geometric properties of the targeted density. However, there also exists hybrid mechanisms, e.g., the O-PMC framework which implements resampling and also incorporates geometric information [23].

### *1.1. Contribution within the state of the art*

In this paper, we propose the gradient-based adaptive multiple importance sampling (GRAMIS) method, which falls within the layered family of AIS algorithms. Its main feature is the exploitation of geometric information about the target by incorporating an optimization approach. It has been shown that geometric-based optimization mechanisms improve the convergence rate in MCMC algorithms (see the review in [28]) and in AIS methods (e.g., [23]). In the context of MCMC, the methods in [29, 30, 31] are called Metropolis adjusted Langevin algorithms (MALA). The Langevin-based adaptation included in their MCMC proposal updates reads as a noisy gradient descent (called drift term) that favors the exploration of the target in areas with larger probability mass, resulting in a larger acceptance probability in the Metropolis-Hastings (MH) step. Preconditioning can be added for a further improvement of the exploration. To do so, local curvature information about the target density is used to build a matrix scaling the drift term. Fisher metric [32], Hessian matrix [33, 34, 35], or a tractable approximation of it [36, 37, 38, 39] have been considered for that purpose. Within AIS, the algorithms in [40, 41] adapt the location parameters via several steps of the unadjusted Langevin algorithm (ULA) [42]. GRAMIS also connects with the Stein variational gradient descent (SVGD) algorithm [43] in the spirit of enhancing the exploratory behavior of the adaptive algorithm. One difference is that SVGD is an MCMC algorithm, while GRAMIS belongs to the family of layered AIS algorithms. Thus, compared to SVGD,

GRAMIS requires less stringent convergence guarantees in the adaptive process for its IS estimators to be consistent. Another difference is that the repulsion term in SVGD builds upon reproducing kernel Hilbert space (RKHS) arguments, while in GRAMIS we justify the repulsion by connecting it to Poisson fields, which also translates into a different adaptive scheme.

A limitation that is present in most AIS algorithms is the lack of adaptation of the scale parameters of the proposals, e.g., the covariance matrices in case of Gaussian proposals. However, suitable scale parameters are essential for an adequate performance of the AIS algorithms, and the alternative to their adaptation is setting them a priori, which is in general a difficult task. The inefficiency of AIS algorithms that do not implement adaptation in the scale parameters is particularly damaging in high dimensions and where the target presents strong correlations that are unknown a priori. A covariance adaptation has been explored via robust moment-matching mechanisms in [44, 45], second-order information in [41, 23], and sample autocorrelation [40]. The proposed GRAMIS algorithm implements an adaptation of the covariance by using second-order information (when it yields to a definite positive matrix). In particular, the covariance adaptation of each proposal is adapted by using the Hessian of the logarithm of the target, evaluated at the updated location parameter. The second-order information is also used to pre-condition the gradient in the adaptation of the location parameters.

Another limitation in AIS algorithms is the lack of cooperation (or insufficient cooperation) between the multiple proposals at the adaptation stage. Some of the few algorithms that implement a type of cooperation can be found in [17] through a probabilistic clustering of all samples, and in [20] through a resampling that use the deterministic mixture weights. In the paper, we implement an explicit repulsion between the proposals during the adaptation stage in order to improve the exploration of the algorithm.

Finally, GRAMIS implements the balance-heuristic estimator (also called deterministic mixture) importance weights [46, 47, 48], which have been shown a theoretical superiority (in the unnormalized IS estimators) [5] and a superior performance in

other types of AIS algorithms (e.g., in [49, 20, 23]).<sup>1</sup>

### 1.2. Notation

We denote by  $\|\mathbf{x}\| = \sqrt{\mathbf{x}^\top \mathbf{x}}$  the Euclidean norm of  $\mathbf{x} \in \mathbb{R}^d$ , where  $\top$  is the transpose operation and  $\mathbb{R}^d$  is the  $d$ -dimensional Euclidean space.  $\nabla$  and  $\nabla^2$  denote the first and second differential operators in  $\mathbf{x}$ , i.e., resulting in the gradient vector and the Hessian matrix, respectively. The operator  $\Sigma \succ 0$  is used to describe  $\Sigma$  as a positive definite matrix.  $\mathbf{Id}_d$  is the identity matrix of dimension  $d$ . Bold symbols are used for vectors and matrices. We use the  $p$  notation for probability density functions (pdf) only when it is unambiguous.

### 1.3. Structure of the paper

The rest of the paper is structured as follows. Section 2 introduces the problem and relevant background. In Section 3, we describe the GRAMIS algorithm. We provide numerical examples in Section 4. Section 5 closes the paper with some conclusion.

## 2. Background in importance sampling

We motivate importance sampling (IS) by considering the Bayesian inference setup, where the goal is characterizing the posterior distribution

$$\tilde{\pi}(\mathbf{x}|\mathbf{y}) = \frac{\ell(\mathbf{y}|\mathbf{x})\pi_0(\mathbf{x})}{Z(\mathbf{y})}, \tag{1}$$

---

<sup>1</sup>The GRAMIS algorithm builds upon the GAPIS algorithm, which was presented in the conference paper [50]. In GRAMIS, on top of updating the location and scale parameters of the proposal parameters by incorporating a repulsion term and using the first and second-order information of the target, we go beyond our preliminary work in the three following ways. First, we pre-condition the gradient steps in the adaptation of the location parameters by using second-order information. In this way, we have a much more efficient adaptation, and we reduce a hyper-parameter w.r.t. the GAPIS algorithm. Second, we find a strong connection with Poisson fields, which supports the theoretical justification of the algorithm and opens the door for further analysis and methodological improvements. Finally, based on this connection, we crucially modify the repulsion term between proposals (e.g., exponentiating the distance between proposal to the dimension of the latent space), which allows the method to scale in dimensions.

where  $\mathbf{x} \in \mathbb{R}^{d_x}$  is the variable associated to the r.v.  $\mathbf{X}$  of the vector of unknowns to be estimated;  $\mathbf{y} \in \mathbb{R}^{d_y}$  represents the available data;  $\ell(\mathbf{y}|\mathbf{x})$  is the likelihood function; and  $\pi_0(\mathbf{x})$  is the prior distribution.

We consider the problem where one must compute the integral

$$I = \int h(\mathbf{x})\tilde{\pi}(\mathbf{x}|\mathbf{y})d\mathbf{x} = \frac{1}{Z(\mathbf{y})} \int h(\mathbf{x})\pi(\mathbf{x}|\mathbf{y})d\mathbf{x}, \quad (2)$$

where  $h$  is any integrable function w.r.t.  $\tilde{\pi}(\mathbf{x}|\mathbf{y})$ . Such problem can arise for instance in the field of Bayesian learning, when  $\mathbf{y}$  gathers the available data to train a model described by vector  $\mathbf{x}$  [51, 52].

In many cases of interest, the integral in Eq. (2) does not have an analytic form. This happens very often in Bayesian inference when the marginal likelihood,  $Z(\mathbf{y}) \triangleq \int \pi(\mathbf{x}|\mathbf{y})d\mathbf{x}$ , is intractable. In these cases, one has access only to the non-negative function  $\pi(\mathbf{x}|\mathbf{y}) \triangleq \ell(\mathbf{y}|\mathbf{x})\pi_0(\mathbf{x}) = Z(\mathbf{y})\tilde{\pi}(\mathbf{x}|\mathbf{y})$ . In the rest of the paper, we alleviate the notation by denoting  $Z$ ,  $\pi(\mathbf{x})$ , and  $\tilde{\pi}(\mathbf{x})$ , i.e., dropping  $\mathbf{y}$  from the notation, since we aim at targeting more generic setups beyond the Bayesian inference problem.

### 2.1. Importance sampling

Importance sampling (IS) is one of the main Monte Carlo methodologies for the approximation of distributions and related integrals. In IS,  $K$  samples  $\mathbf{x}_k$  are simulated from an alternative distribution  $q(\mathbf{x})$ , called proposal, and receive an importance weight  $w_k$ . The procedure comprises two basic steps:

1. **Sampling.**  $K$  samples are drawn as

$$\mathbf{x}_k \sim q(\mathbf{x}), \quad k = 1, \dots, K.$$

2. **Weighting.** Each sample is associated to an importance sampling

$$w_k = \frac{\pi(\mathbf{x}_k)}{q(\mathbf{x}_k)}, \quad k = 1, \dots, K.$$

The resulting sets of samples  $\{\mathbf{x}_k\}_{k=1}^K$  and weights  $\{w_k\}_{k=1}^K$  are used in order to produce estimators that approximate  $I$  in Eq. (2). When  $Z$  is available, it is possible to produce the unnormalized IS (UIS) estimator

$$\hat{I} = \frac{1}{KZ} \sum_{k=1}^K w_k h(\mathbf{x}_k). \quad (3)$$

Notation	Description
$\mathbf{x} \in \mathbb{R}^{d_x}$	variable associated to the r.v. $\mathbf{X}$ of interest
$\mathbf{y} \in \mathbb{R}^{d_y}$	vector of data
$\tilde{\pi}(\mathbf{x}) \triangleq \tilde{\pi}(\mathbf{x} \mathbf{y})$	target pdf
$\pi(\mathbf{x}) \triangleq \pi(\mathbf{x} \mathbf{y}) \triangleq \ell(\mathbf{y} \mathbf{x})\pi_0(\mathbf{x}) = Z(\mathbf{y})\tilde{\pi}(\mathbf{x} \mathbf{y})$	unnormalized target density function
$Z \triangleq Z(\mathbf{y})$	normalizing constant
$h(\mathbf{x})$	test function
$\ell(\mathbf{y} \mathbf{x})$	likelihood function
$\pi_0(\mathbf{x})$	prior pdf
$\hat{I}$	unnormalized IS (UIS) estimator
$\tilde{I}$	self-normalized IS (SNIS) estimator
$N$	number of proposals
$K$	number of samples per proposal
$T$	number of iterations
$q_n^{(t)}(\mathbf{x}; \boldsymbol{\mu}_n^{(t)}, \boldsymbol{\Sigma}_n^{(t)}, \boldsymbol{\xi}_n)$	$n$ -th proposal pdf at $t$ -th iteration
$\boldsymbol{\mu}_n^{(t)}$	location parameter of the $n$ -th proposal pdf at $t$ -th iteration
$\boldsymbol{\Sigma}_n^{(t)}$	scale parameter of the $n$ -th proposal pdf at $t$ -th iteration
$\boldsymbol{\xi}_n$	non-adapted parameters of the $n$ -th proposal pdf
$\mathbf{x}_{n,k}^{(t)}$	$k$ -th sample from the $n$ -th proposal at $t$ -th iteration
$w_{n,k}^{(t)}$	importance weight associated to $\mathbf{x}_{n,k}^{(t)}$

Table 1: Summary of notation of the main variables, functions, constants, and parameters. Some functions are explicit for a generic Bayesian inference problem.

If  $Z$  is not available, the alternative is to use the self-normalized IS (SNIS) estimator,

$$\tilde{I} = \sum_{k=1}^K \bar{w}_k h(\mathbf{x}_k), \quad (4)$$

where  $\bar{w}_k = w_k / \sum_{j=1}^K w_j$ ,  $k = 1, \dots, K$  are the importance weights.

The UIS estimator is unbiased and consistent. The SNIS estimator is consistent and has a bias which vanishes at a faster rate than the variance when  $K$  grows. The optimal proposal of the UIS estimator is  $q(\mathbf{x}) \propto |h(\mathbf{x})|\pi(\mathbf{x})$  [1, 2], while the optimal proposal of the SNIS estimator is (approximately)  $q(\mathbf{x}) \propto |h(\mathbf{x})|\pi(\mathbf{x})$  [3].

## 2.2. Multiple importance sampling

One of the most common strategies is to use several proposals,  $\{q_n(\mathbf{x})\}_{n=1}^N$  [53, 46, 3]. The last years have witnessed and increased of attention in MIS [54, 55, 56, 57, 58] (see a generic framework with theoretical analysis in [5]). It has been shown that several weighting and sampling schemes are possible, i.e., that lead to consistent UIS and SNIS estimators [5]. We consider the simplified example where we simulate  $K = N$  samples from the set of proposals. Then, one possibility is to simulate exactly one sample per proposal as  $\mathbf{x}_n \sim q_n(\mathbf{x})$ ,  $n = 1, \dots, N$ . Then, two popular weighting approaches are the standard MIS (s-MIS),

$$w_n = \frac{\pi(\mathbf{x}_n)}{q_n(\mathbf{x}_n)}, \quad n = 1, \dots, N,$$

and the deterministic mixture MIS (DM-MIS),

$$w_n = \frac{\pi(\mathbf{x}_n)}{\frac{1}{N} \sum_{j=1}^N q_j(\mathbf{x}_n)} = \frac{\pi(\mathbf{x}_n)}{\frac{1}{N} \sum_{j=1}^N q_j(\mathbf{x}_n)}, \quad n = 1, \dots, N.$$

In [5], it is proved that DM-MIS weights provide an UIS estimator with less variance compared to s-MIS, for any function  $h$ , any target  $\tilde{\pi}$ , and any set of proposals.

## 2.3. Optimization-based samplers

The performance of sampling algorithms depends greatly on the choice of the proposal distribution. Proposals parametrized with static parameters are easier to implement and manipulate, as they require minimal self tuning, but this simplicity comes at the price of a suboptimal (since not adaptative) target exploration. To cope with this



issue, several methods have been proposed to iteratively update the proposal, along the sampling algorithm iterations, so as to improve and accelerate the target exploration. The most common technique is to resort to a Langevin-based approach, where gradient descent steps (assuming differentiability of  $\log \pi$ ) are performed to adapt the proposal mean (i.e., location). The discretization of the Langevin dynamics leads to the unadjusted Langevin algorithm (ULA) [42], which can also be viewed as a gradient descent algorithm perturbed with an independent and identically distributed (i.i.d.) stochastic error. The convergence of ULA is discussed in [59, 42]. However, in most situations, the stationary distribution of the samples produced by ULA differs from the target  $\pi$  [60], due to the discretization of the Langevin dynamic. MALA [29] tackles this issue by introducing an MH strategy, hence guaranteeing ergodic convergence to the sought target law. Accelerated variants of MALA have been investigated, based on preconditioning techniques to account for more information (e.g., curvature) about the target [36, 32, 61, 35, 62, 29, 63]. For instance, the Newton MH strategy [61, 35] consists in combining an MH procedure with a stochastic Newton update involving the inverse (or an approximation of it, when undefined or too complex) of the Hessian matrix of  $\log \pi(\mathbf{x})$ . This approach will serve as starting point for introducing a proposal adaptation within our novel approach GRAMIS.

### 3. The GRAMIS algorithm

We now describe GRAMIS, the proposed AIS algorithm, in Table 2. The algorithm runs over  $T$  iterations, adapting  $N$  proposals, and simulating  $K$  sampler per proposal and iteration. At the  $t$ -th iteration, we adapt the location parameter,  $\boldsymbol{\mu}_n^{(t)}$ , and the scale parameter,  $\boldsymbol{\Sigma}_n^{(t)}$ , of each proposal, with  $n = 1, \dots, N$ . The non-adapted parameters are denoted as  $\boldsymbol{\xi}_n$  (for instance, the degrees of freedom when using Student's t-proposals).

First, the location parameters are updated in (9) following a gradient step that includes an optimized stepsize  $\theta_n^{(t-1)}$ , and first and second-order information of the log-target at the previous location parameter  $\boldsymbol{\mu}_n^{(t-1)}$  of each proposal (see Section 3.1 for more details). A repulsion term between each pair of proposals (i.e., between  $j$ -th and  $i$ -th proposals),  $\mathbf{r}_{i,j}^{(t-1)}$ , is introduced. This repulsion force which inversely proportional

to the (Euclidean) distance  $\|\mathbf{d}_{i,j}\| = \|\boldsymbol{\mu}_i^{(t-1)} - \boldsymbol{\mu}_j^{(t-1)}\|$ . More practical information about the choice of repulsion strategy can be found in Section 3.3. Second, the scale parameters are updated by using second-order information of the log-target (see Section 3.2). Third,  $K$  samples are simulated from each proposal. The importance weights are computed in Eq. (11). Note that we implement the DM-MIS weights as in Eq. (5), which is guaranteed to reduce the variance of the UIS estimators [5, Theorems 1 and 2]. GRAMIS returns  $KNT$  weighted samples that can be used to estimate both  $I$  and  $Z$  (in the case it is unknown). The simplest version of those estimators is given below.

- UIS estimator:

$$\hat{I} = \frac{1}{KTNZ} \sum_{t=1}^T \sum_{n=1}^N \sum_{k=1}^K w_{n,k}^{(t)} h(\mathbf{x}_{n,k}^{(t)}). \quad (5)$$

- SNIS estimator:

$$\tilde{I} = \sum_{t=1}^T \sum_{n=1}^N \sum_{k=1}^K \tilde{w}_{n,k}^{(t)} h(\mathbf{x}_{n,k}^{(t)}), \quad (6)$$

where

$$\tilde{w}_{n,k}^{(t)} = \frac{w_{n,k}^{(t)} h(\mathbf{x}_{n,k}^{(t)})}{\sum_{t=1}^T \sum_{n=1}^N \sum_{k=1}^K w_{n,k}^{(t)}} \quad (7)$$

are the re-normalized weights.

- estimator of  $Z$ :

$$\hat{Z} = \frac{1}{KTNZ} \sum_{t=1}^T \sum_{n=1}^N \sum_{k=1}^K w_{n,k}^{(t)}. \quad (8)$$

### 3.1. Adaptation of location parameters

The location parameters are adapted as in Eq. (9). The adaptation process implements a Newton ascent on  $\log \pi$ . The gradient of the log target is evaluated at the previous location parameter and pre-conditioned by  $\boldsymbol{\Sigma}_n^{(t-1)}$  where  $\boldsymbol{\Sigma}_n^{(t-1)}$  is the same that we use in the previous section in order to update the covariance. We furthermore introduced  $\theta_n^{(t-1)} \in (0, 1]$ , which is a stepsize tuned according to a backtracking scheme in order to avoid the degeneracy of the Newton iteration, and thus of our adaptation scheme, for non log-concave distributions. Starting with unit stepsize value, we reduce it by factor 1/2 until the condition below is met:

$$\pi \left( \boldsymbol{\mu}_n^{(t-1)} + \theta_n^{(t-1)} \boldsymbol{\Sigma}_n^{(t-1)} \nabla \log \left( \pi(\boldsymbol{\mu}_n^{(t-1)}) \right) \right) \geq \pi \left( \boldsymbol{\mu}_n^{(t-1)} \right), \quad (12)$$

Table 2: GRAMIS.

1. **[Initialization]:** Initialize the proposal means  $\boldsymbol{\mu}_n^{(0)}$  and the non-adapted parameters  $\boldsymbol{\xi}_n$ . Compute the scale parameter matrix  $\boldsymbol{\Sigma}_n^{(0)}$  using (13).

2. **[For  $t = 1$  to  $T$ ]:**

(a) **Mean adaptation:**

- i. Compute the stepsize  $\theta_n^{(t-1)}$  using the backtracking procedure so as to satisfy (12).
- ii. The mean of the  $n$ -th proposal is adapted as

$$\boldsymbol{\mu}_n^{(t)} = \boldsymbol{\mu}_n^{(t-1)} + \theta_n^{(t-1)} \boldsymbol{\Sigma}_n^{(t-1)} \nabla \log \left( \pi(\boldsymbol{\mu}_n^{(t-1)}) \right) + \sum_{j=1, j \neq n}^N \mathbf{r}_{n,j}^{(t-1)}, \quad (9)$$

with

$$\mathbf{r}_{n,j}^{(t-1)} = G_t \frac{m_n m_j}{\|\mathbf{d}_{n,j}^{(t-1)}\|_{d_t}} \mathbf{d}_{n,j}^{(t-1)}, \quad (10)$$

where  $\|\cdot\|$  represents the norm operator,  $\mathbf{d}_{n,j}^{(t-1)} = \boldsymbol{\mu}_n^{(t-1)} - \boldsymbol{\mu}_j^{(t-1)}$ , and  $m_n, m_j > 0$  are two positive terms that depend on the  $n$ -th and  $j$ -th proposals respectively.

(b) **Covariance adaptation:** The covariance matrix of the  $n$ -th proposal  $\boldsymbol{\Sigma}_n^{(t)}$  is adapted using (13).

(c) **Sampling steps:**

- i. Draw  $K$  independent samples from each proposal, i.e.,  $\mathbf{x}_{n,k}^{(t)} \sim q_n^{(t)}(\mathbf{x}; \boldsymbol{\mu}_n^{(t)}, \boldsymbol{\Sigma}_n^{(t)}, \boldsymbol{\xi}_n)$  for  $k = 1, \dots, K$  and  $n = 1, \dots, N$ .
- ii. Compute the importance weights,

$$w_{n,k}^{(t)} = \frac{\pi(\mathbf{x}_{n,k}^{(t)})}{\frac{1}{N} \sum_{j=1}^N d_j^{(t)}(\mathbf{x}_{n,k}^{(t)}; \boldsymbol{\mu}_n^{(t)}, \boldsymbol{\Sigma}_n^{(t)})}, \quad (11)$$

for  $n = 1, \dots, N$ , and  $k = 1, \dots, K$ .

3. **[Output]:** Return the pairs  $\{\mathbf{x}_{n,k}^{(t)}, w_{n,k}^{(t)}\}$ , for  $n = 1, \dots, N$ ,  $k = 1, \dots, K$ , and  $t = 1, \dots, T$ .

i.e., the repulsion term is not considered in the backtracking scheme.

The update in Eq. (9) also incorporates an innovative repulsion term among proposals. The purpose is to efficiently explore the space in a cooperative manner. This repulsion term admits several interpretations. It can be seen as an over-spreading of the mixture proposal, i.e., a safer choice of mixture that will overweight the tails of the target [47]. Also, it can be interpreted as a negative coupling among proposals. It shares connections with MCMC algorithms that implement interacting parallel chains, with a

similar spirit as it is done in MCMC [64, 65]. In Section 3.3, we discuss the practical repulsion schemes, and the rationale of the adaptation is discussed in Section 3.4.

### 3.2. Adaptation of scale parameters

We implement a Newton-based strategy, inherited from the literature of optimization [66], to exploit the Hessian of  $\log \pi$  in the update of the scale parameter. In general scenarios, the convexity of  $-\log \pi$  is not ensured, and numerical issues might arise when computing the inverse of its Hessian. We thus propose to introduce a safe rule in our adaptation method, so that

$$\Sigma_n^{(t)} = \begin{cases} \left(-\nabla^2 \log \pi(\boldsymbol{\mu}_n^{(t)})\right)^{-1}, & \text{if } \nabla^2 \log \pi(\boldsymbol{\mu}_n^{(t)}) \succ 0, \\ \Sigma_n^{(t-1)}, & \text{otherwise.} \end{cases} \quad (13)$$

The scaling matrix thus incorporates second order information on the target, whenever this yields to a definite positive matrix. Otherwise, it inherits the covariance of the proposal of the previous iteration, where  $\Sigma_n^{(0)}$  is set to a predefined default value (typically, a scalar times the identity matrix).

### 3.3. Design of the repulsion scheme

Our repulsion term is parameterized by a common time-dependent constant  $G_t$  and a proposal-dependent constant,  $m_n$ , for each  $n = 1, \dots, N$ . By construction, Eq. (10) implies that the repulsion term vanishes whenever the proposals get further away (in Euclidean distance).

The interpretation of the functional form in Eq. (10) is discussed in the next section. The simpler choice is to keep  $m_n = 1$ , for all  $n = 1, \dots, N$ , and to fix the common term  $G_t$  to be constant over the iterations, i.e.,  $G_t = G$ . In this case, the repulsion never vanishes with the consequence of leading to a potential equilibrium positioning of the proposals in such a way that the interpreted mixture proposal,  $\tilde{q}^{(t)}(\mathbf{x}) \triangleq \frac{1}{N} \sum_{n=1}^N q_n^{(t)}(\mathbf{x}; \boldsymbol{\mu}_n^{(t)}, \Sigma_n^{(t)}, \boldsymbol{\xi}_n)$  would overweight the tails of the target distribution. In this case, it is not guaranteed that the proposal adaptation converges in finite  $t$ . An alternative is to reduce the repulsion term in such a way that  $r_{n,j}^{(t)} \rightarrow 0$  when  $t \rightarrow \infty$ . A natural choice is a decaying term in the form of

$$G_t = \exp(-\beta t), \quad \beta > 0. \quad (14)$$

In such case, if the Newton scheme converges to a local maximum for each proposal, the whole mixture approximation would converge to a mixture of local Laplace approximations. The choice of  $\beta$  can be easily set depending on the repulsion strength desired in the last iteration, e.g., a 1% of attenuation in the last iteration leads to  $\beta = \frac{-\log(0.01)}{T-1}$ . It is also possible to set the repulsion term to zero in the last iteration, so a final set of samples can be simulated.

### 3.4. Interpretation of the repulsion term

We can interpret the repulsion term of Eq. (10) in general physical terms. The following discussion uses the particle interpretation of the repulsion term mentioned in [50], formalizing it using the notion of Poisson fields [67]. In what follows, we first introduce the notion of Poisson fields in Section 3.4.1 and then connect this to our algorithm in Section 3.4.2.

#### 3.4.1. Poisson fields

This section summarizes Poisson fields as in [67]. Let  $\rho(\mathbf{v}) : \mathbb{R}^{d_x} \rightarrow \mathbb{R}$  be a *source function* with a compact support. In this setting, the Poisson equation is defined as

$$\nabla^2 \varphi(\mathbf{v}) = -\rho(\mathbf{v}), \quad (15)$$

where  $\varphi(\mathbf{v}) : \mathbb{R}^{d_x} \rightarrow \mathbb{R}$  is called *the potential function*. The gradient  $\nabla \varphi(\mathbf{v})$  is usually interpreted as a *field* and may have a physical meaning analogous to the low dimensional cases. For example, writing  $\mathbf{E}(\mathbf{v}) = -\nabla \varphi(\mathbf{v})$  converts the Poisson equation into  $\nabla \cdot \mathbf{E} = \rho$  which is Gauss' law in physics [67]. In this case,  $\mathbf{E}$  would be a  $d_x$ -dimensional generalization of the 3-dimensional electric field. Similarly  $\rho$  would be a  $d_x$  dimensional analogue of the electrical charge density.

Equation (15) has a unique solution (with extra regularity conditions [67, Lemma 1]) given as

$$\varphi(\mathbf{u}) = \int G(\mathbf{u}, \mathbf{v}) \rho(\mathbf{v}) d\mathbf{v},$$

where

$$G(\mathbf{u}, \mathbf{v}) = \frac{1}{(d_x - 2)S_{d_x-1}(1)} \frac{1}{\|\mathbf{u} - \mathbf{v}\|^{d_x-2}},$$

and

$$S_{d_x-1}(1) = \frac{2\pi^{d_x/2}}{\Gamma(d_x/2)}$$

is a constant equals to the surface area of the unit  $(d_x - 1)$  sphere (note that the particle interpretation is then valid for in  $\mathbb{R}^{d_x}$  for  $d_x \geq 3$ ). Then, the negative gradient field is called a *Poisson field* [67] given as

$$-\nabla\varphi(\mathbf{u}) = - \int \nabla_{\mathbf{u}}G(\mathbf{u}, \mathbf{v})\rho(\mathbf{v})d\mathbf{v},$$

where

$$\nabla_{\mathbf{u}}G(\mathbf{u}, \mathbf{v}) = - \frac{1}{S_{d_x-1}(1)} \frac{\mathbf{u} - \mathbf{v}}{\|\mathbf{u} - \mathbf{v}\|^{d_x}}.$$

The property of the Poisson field  $-\nabla\varphi(\mathbf{u})$  is that, it creates a field that moves a particle away from sources  $\rho(\mathbf{v})$ . In the case of a single source (i.e.,  $\rho(\mathbf{v})$  is a Dirac), this quantity would create a repulsive effect w.r.t. this particle. In what follows, we will interpret the repulsion term in our GRAMIS scheme in terms of an *empirical* Poisson field where the source distribution will coincide with the GRAMIS proposals in the previous iteration.

### 3.4.2. Repulsion term as an empirical Poisson field

In this section, for the ease of presentation, we consider a simplified version of the mean adaptation defined in (9). In particular, we consider a fixed, scalar step-size  $\theta_n^{(t-1)} = \gamma$  for all  $n = 1, \dots, N$  and  $t = 1, \dots, T$  and  $\Sigma_n^{(t-1)} = I_{d_x}$ , where  $I_{d_x}$  is an identity matrix. In this case, the update rule takes this form:

$$\boldsymbol{\mu}_n^{(t)} = \boldsymbol{\mu}_n^{(t-1)} + \gamma \nabla \log \left( \pi(\boldsymbol{\mu}_n^{(t-1)}) \right) + \sum_{j=1, j \neq n}^N \mathbf{r}_{n,j}^{(t-1)}, \quad (16)$$

where  $\gamma > 0$  is a scalar step-size. For the repulsion term, we also provide some choices that would lead to clarification of the role of the repulsion term. In particular, we set

$$G_t = \frac{\gamma}{(N-1)S_{d_x-1}(1)} \quad (17)$$

for all  $t = 1, \dots, T$  and  $m_n = 1$  for all  $n = 1, \dots, N$ .<sup>2</sup> As a consequence, the repulsion term takes the following form:

$$\mathbf{r}_{n,j}^{(t-1)} = \frac{\gamma}{(N-1)S_{d_x-1}(1)} \frac{1}{\|\mathbf{d}_{n,j}^{(t-1)}\|^{d_x}} \mathbf{d}_{n,j}^{(t-1)}. \quad (18)$$

where  $\mathbf{d}_{n,j}^{(t-1)} = \boldsymbol{\mu}_n^{(t-1)} - \boldsymbol{\mu}_j^{(t-1)}$ . Our aim is now to interpret the repulsion term in (16) as a Poisson field which creates a repulsive effect for location parameter  $n$  and pushes it away from the other location parameters  $\boldsymbol{\mu}_j^{(t-1)}$ , with  $j \in \{1, \dots, N\} \setminus n$ .

Our first step is to interpret the last term in (16) as an empirical estimate of an integral. In order to do so, recall that with the choice of (18), the last term of (16) can be written as

$$\sum_{j=1, j \neq n}^N \mathbf{r}_{n,j}^{(t-1)} = \frac{\gamma}{(N-1)} \sum_{j=1, j \neq n} \frac{1}{S_{d_x-1}(1)} \frac{1}{\|\mathbf{d}_{n,j}^{(t-1)}\|^{d_x}} \mathbf{d}_{n,j}^{(t-1)}, \quad (19)$$

where  $\mathbf{d}_{n,j}^{(t-1)} = \boldsymbol{\mu}_n^{(t-1)} - \boldsymbol{\mu}_j^{(t-1)}$ . The sum in the r.h.s. above can now be interpreted as an integral approximation. More precisely, let us consider the empirical measure constructed by the sequence  $\{\boldsymbol{\mu}_n^{(t-1)}\}_{n=1}^N$  given by

$$\rho_{t-1,n}^N(\mathbf{d}\mathbf{u}) = \frac{1}{N-1} \sum_{j=1, j \neq n}^N \delta_{\boldsymbol{\mu}_j^{(t-1)}}(\mathbf{d}\mathbf{u}),$$

which is defined for the update of  $\boldsymbol{\mu}_n^{(t-1)}$ . Using this empirical measure, we can interpret the repulsion term in (19) as

$$\frac{\gamma}{(N-1)} \sum_{j=1, j \neq n} \frac{1}{S_{d_x-1}(1)} \frac{1}{\|\mathbf{d}_{n,j}^{(t-1)}\|^{d_x}} \mathbf{d}_{n,j}^{(t-1)} = \gamma \int g(\boldsymbol{\mu}_n^{(t-1)}, \mathbf{v}) \rho_{t-1,n}^N(\mathbf{d}\mathbf{v}), \quad (20)$$

with

$$g(\mathbf{u}, \mathbf{v}) = \frac{1}{S_{d_x-1}(1) \|\mathbf{u} - \mathbf{v}\|^{d_x}} (\mathbf{u} - \mathbf{v}). \quad (21)$$

Finally, embedding this into our update rule (16), we obtain

$$\boldsymbol{\mu}_n^{(t)} = \boldsymbol{\mu}_n^{(t-1)} + \gamma \nabla \log \left( \pi(\boldsymbol{\mu}_n^{(t-1)}) \right) + \gamma \int g(\boldsymbol{\mu}_n^{(t-1)}, \mathbf{v}) \rho_{t-1,n}^N(\mathbf{d}\mathbf{v}), \quad (22)$$

---

<sup>2</sup>Note that this is not restrictive, as any choice of  $G_t$  can be multiplied and divided by  $\frac{(N-1)S_{d_x-1}(1)}{\gamma}$  to define  $\tilde{G}_t = \frac{(N-1)S_{d_x-1}(1)}{\gamma} G_t$ . These constants can also be used to choose the parameters  $m_n$ .

This implies that, if we set  $g(\mathbf{u}, \mathbf{v}) = -\nabla_{\mathbf{u}}G(\mathbf{u}, \mathbf{v})$ , where

$$G(\mathbf{u}, \mathbf{v}) = \frac{1}{(d_x - 2)\mathcal{S}_{d_x-1}(1)} \frac{1}{\|\mathbf{u} - \mathbf{v}\|^{d_x-2}},$$

then the last term in (22) can be interpreted as a gradient [67], i.e.,

$$-\nabla \varphi_{t-1}^N(\mathbf{u}) = -\int \nabla_{\mathbf{u}}G(\mathbf{u}, \mathbf{v})\rho_{t-1,n}^N(d\mathbf{v}).$$

Finally, going back to the update of the location parameters, we can then rewrite (16) as

$$\boldsymbol{\mu}_n^{(t)} = \boldsymbol{\mu}_n^{(t-1)} + \gamma \nabla \log \left( \pi(\boldsymbol{\mu}_n^{(t-1)}) \right) - \gamma \nabla \varphi_{t-1}^N(\boldsymbol{\mu}_n^{(t-1)}), \quad (23)$$

where

$$-\nabla \varphi_{t-1}^N(\boldsymbol{\mu}_n^{(t-1)}) = \frac{1}{N-1} \sum_{j=1, j \neq n}^N \frac{\Gamma(d_x/2)}{2\pi^{d_x/2}} \frac{\mathbf{d}_{n,j}^{(t-1)}}{\|\mathbf{d}_{n,j}^{(t-1)}\|^{d_x}}.$$

One can clearly see that  $-\nabla \varphi_{t-1}^N(\mathbf{u})$  is a *Poisson field* as defined in Section 3.4.1 with  $\rho_{t-1,n}^N$  as the *empirical source distribution*. In other words, the term  $-\nabla \varphi_{t-1}^N(\mathbf{u})$ , as a field, would push a location parameter away from all others in that constructs the empirical distribution  $\rho_{t-1,n}^N$ . The update (23) has two effects by pushing the  $n$ -th location parameter to maximize  $\pi$  due to the effect of  $\nabla \log \left( \pi(\boldsymbol{\mu}_n^{(t-1)}) \right)$ , and also by staying away from all other location parameters by the repulsion term  $-\nabla \varphi_{t-1}^N(\boldsymbol{\mu}_n^{(t-1)})$ , i.e. *the empirical Poisson field*. More explicitly, Eq. (23) describes the precise balance achieved by the repulsion term. The term  $\nabla \log \pi(\cdot)$  in (23) pushes the means towards the maximum-a-posteriori (MAP) estimate, which, if implemented alone, would cause all means to converge to the MAP (e.g., in settings where target has a single maximum, i.e., MAP is uniquely defined). The repulsion term creates a potential for means to stay away from each other. More precisely, the last term in (23) pushes means away from *sources*. In our case, sources for the adaptation of the  $n$ -th location parameter are the other location parameters  $\{\boldsymbol{\mu}_j^{(t-1)}\}_{j=1, j \neq n}^N$ . In other words, the addition of the term  $-\nabla \varphi_{t-1}^N(\boldsymbol{\mu}_n^{(t-1)})$  to the gradient flow above creates a repulsive effect, pushing the updated mean away from the location parameters, which effectively spreads out the components in the mixture proposal. This interpretation also holds when we introduce back  $m_n, n = 1, \dots, N$  terms, effectively determining the strength of the repulsion for a



particular mean. From this viewpoint,  $G_t$  can also be seen as the adaptive weight that determines whether the repulsion term should be more or less active. High values of  $G_t$  might be useful in the initial exploration phase.

## 4. Numerical experiments

### 4.1. Ablation study with Gaussian mixtures

Let us consider a generic mixture of bivariate Gaussian pdfs as

$$(\forall \mathbf{x} \in \mathbb{R}^2) \quad \tilde{\pi}(\mathbf{x}) = \frac{1}{L} \sum_{\ell=1}^L \mathcal{N}(\mathbf{x}; \boldsymbol{\gamma}_\ell, \boldsymbol{\Sigma}_\ell). \quad (24)$$

We start considering a toy example with  $L = 2$  components to better understand the behavior of GRAMIS. In this case, the means are  $\boldsymbol{\gamma}_1 = [-5, -5]^\top$  and  $\boldsymbol{\gamma}_2 = [6, 4]^\top$ , and the covariance are  $\boldsymbol{\Sigma}_1 = [0.25, 0; 0, 0.25]$  and  $\boldsymbol{\Sigma}_2 = [0.52, 0.48; 0.48, 0.52]$ . We run GRAMIS displaying the adaptive behavior of different ablated version of the algorithm. We set  $N = 50$  Gaussian proposals,  $T = 20$  iterations, and  $K = 20$  samples per proposal and iteration. The location parameters of the proposals are randomly initialized in the square  $[1, 6] \times [1, 6]$ .

Figure 1 shows the final location parameters (black dots) and scale parameters (black ellipses) of the proposals at time  $t = 20$  for four ablated versions of GRAMIS. Plot (a) shows the modified GRAMIS without preconditioning matrix in the gradient update (as in [50] with  $\lambda = 0.1$ ) and  $G_t = 0$  (no repulsion). In this setting, the adaptation of the location parameter simplifies into

$$\boldsymbol{\mu}_n^{(t)} = \boldsymbol{\mu}_n^{(t-1)} + \lambda \nabla \log \left( \pi(\boldsymbol{\mu}_n^{(t-1)}) \right). \quad (25)$$

The arbitrary (suboptimal) step-size delays the convergence of the location parameter to the mode. Plot (b) shows GRAMIS with  $G_t = 0$  (no repulsion). The adaptation is effective in recovering one mode, but not in finding the second mode. Plot (c) shows GRAMIS with constant repulsion  $G_t = 0.5$ . The mixture of proposals has *discovered* both modes, but since the repulsion is not decreased, the proposals cannot concentrate around the mode. Plot (d) shows GRAMIS with exponentially decayed repulsion  $G_1 = 0.5$  (see Section 3.3), with the mixture proposal successfully approximating the

target density. We denote this mixture as  $\tilde{q}^{(T)}(\mathbf{x}) = \frac{1}{N} \sum_{n=1}^N q_n^{(T)}(\mathbf{x}; \boldsymbol{\mu}_n^{(T)}, \boldsymbol{\Sigma}_n^{(T)})$ , i.e., the mixture density composed of all the proposals at the last iteration of the algorithm. In all cases, we show also the marginal plots of  $\tilde{q}^{(T)}(\mathbf{x})$  and  $\tilde{\pi}(\mathbf{x})$ .

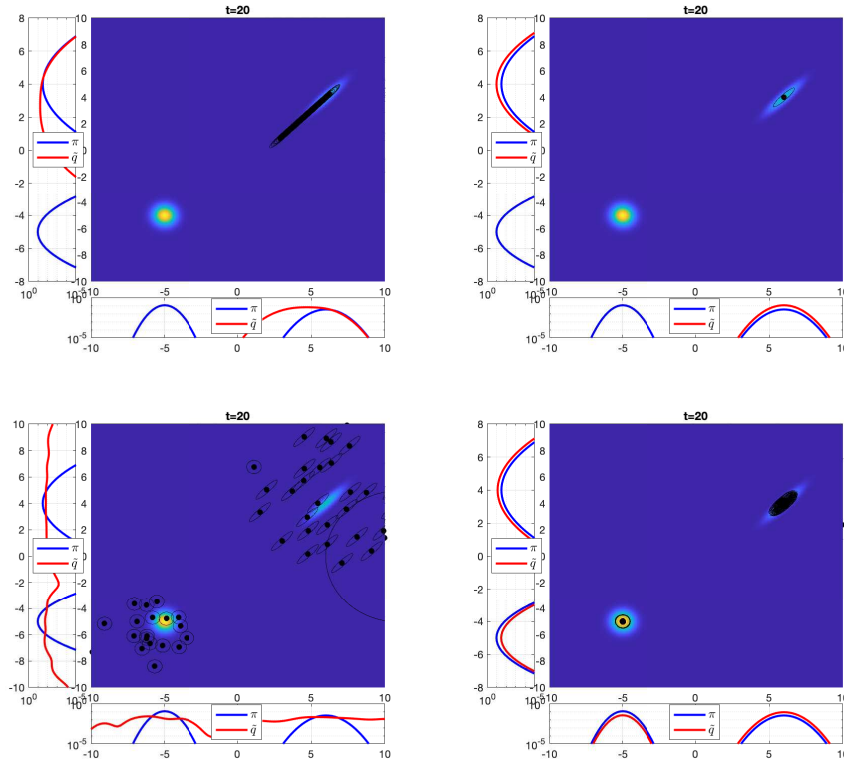


Figure 1: **Toy example.** Final location parameters (black dots) and scale parameters (black ellipses) of the proposals at time  $t = 20$  for four ablated versions of GRAMIS. Upper left: modified GRAMIS without preconditioning matrix in the gradient update (as in [50] with  $\lambda = 0.1$ ) and  $G_t = 0$  (no repulsion). Upper right: GRAMIS with  $G_t = 0$  (no repulsion). Bottom left: GRAMIS with constant repulsion  $G_t = 0.5$ . Bottom right: GRAMIS with exponentially decayed repulsion  $G_1 = 0.5$  (see Section 3.3).

We now extend the setting to  $L = 5$  components, with means  $\boldsymbol{\gamma}_1 = [-10, -10]^\top$ ,  $\boldsymbol{\gamma}_2 = [0, 16]^\top$ ,  $\boldsymbol{\gamma}_3 = [13, 8]^\top$ ,  $\boldsymbol{\gamma}_4 = [-9, 7]^\top$ ,  $\boldsymbol{\gamma}_5 = [14, -4]^\top$ , and covariance matrices  $\boldsymbol{\Sigma}_1 = [5, 2; 2, 5]$ ,  $\boldsymbol{\Sigma}_2 = [2, -1.3; -1.3, 2]$ ,  $\boldsymbol{\Sigma}_3 = [2, 0.8; 0.8, 2]$ ,  $\boldsymbol{\Sigma}_4 = [3, 1.2; 1.2, 0.5]$  and  $\boldsymbol{\Sigma}_5 = [0.2, -0.1; -0.1, 0.2]$ . This target is particularly challenging since it requires the algorithms to discover 5 modes. We aim at estimating the first and second moments,

and the normalizing constant, which are available in a closed form. The proposals are now randomly initialized in the square  $[-15, 15] \times [-15, 15]$ .

Table 3 shows the RMSE in the estimation of  $Z$  and the first and second moments of the target in an ablation study of GRAMIS. In particular, we test four versions of the algorithm, with/without preconditioning matrix in the update of the location parameters and with/without repulsion, i.e., the last column is the GRAMIS algorithm Table 2. In the case without preconditioning matrix, we set  $\gamma = 10^{-1}$ . In the case with repulsion, we use  $G_1 = 0.05$  with exponential decay, otherwise we simply set  $G_1 = 0$  to annihilate the repulsion effect. The MSE results are obtained over 100 independent runs, with estimators using the weighted samples on the half last iterations. It can be seen that the worst results are obtained when no preconditioning and no repulsion are implemented, while the best results are obtained by the full GRAMIS algorithm.

RMSE	No pre-cond./No repulsion			No pre-cond./Repulsion			Pre-cond./No repulsion			Pre-cond./Repulsion		
	$\sigma = 1$	$\sigma = 3$	$\sigma = 5$	$\sigma = 1$	$\sigma = 3$	$\sigma = 5$	$\sigma = 1$	$\sigma = 3$	$\sigma = 5$	$\sigma = 1$	$\sigma = 3$	$\sigma = 5$
$Z$	1.0108	0.0339	0.3152	0.0296	0.0163	0.0291	0.0224	0.0128	0.0192	<b>0.0096</b>	0.0168	0.0264
$E_{\tilde{\pi}}[\mathbf{X}]$	2.7567	1.6222	2.6798	0.7492	0.7253	0.5969	1.4756	0.9557	1.2745	<b>0.7694</b>	0.9097	1.5663
$E_{\tilde{\pi}}[\mathbf{X}^2]$	2.5058	1.7431	2.3161	0.6521	0.7439	0.3422	1.6427	0.8829	1.7884	<b>0.8137</b>	0.8895	1.6063

Table 3: **Gaussians-mixture target in Section 4.1.** RMSE of the IS estimators. We run an ablation study of GRAMIS with/without preconditioning matrix in the update of the location parameters and with/without repulsion. In the case without preconditioning matrix, we set  $\gamma = 10^{-1}$ . In the case with repulsion,  $G_1 = 0.05$  with exponential decay. The MSE results are obtained over 100 independent runs, with estimators using the weighted samples on the half last iterations.

#### 4.2. Generalized Gaussian mixtures

We consider a generalization of the previous study where the target is now a mixture of  $L \geq 1$  bivariate generalized Gaussian pdfs as

$$(\forall \mathbf{x} \in \mathbb{R}^2) \quad \tilde{\pi}(\mathbf{x}) = \sum_{\ell=1}^L \omega_{\ell} \mathcal{GG}(\mathbf{x}; \boldsymbol{\nu}_{\ell}, \boldsymbol{\Sigma}_{\ell}, \eta_{\ell}), \quad (26)$$

where, for every  $\ell \in \{1, \dots, L\}$ ,  $\omega_{\ell}$  are the mixture weights, and  $\boldsymbol{\nu}_{\ell}$ ,  $\boldsymbol{\Sigma}_{\ell}$ , and  $\eta_{\ell}$  are respectively the mean, scale, and shape parameters of each component of the mixture

[68], which is given by

$$\mathcal{GG}(\mathbf{x}; \boldsymbol{\nu}_\ell, \boldsymbol{\Sigma}_\ell, \eta_\ell) = \frac{d_x \Gamma(\frac{d_x}{2})}{\pi^{\frac{d_x}{2}} \Gamma(1 + \frac{d_x}{2\eta_\ell}) 2^{1 + \frac{d_x}{2\eta_\ell}}} |\boldsymbol{\Sigma}_\ell|^{-1/2} \exp\left(-\frac{1}{2} \left((\mathbf{x} - \boldsymbol{\nu}_\ell)^\top \boldsymbol{\Sigma}_\ell^{-1} (\mathbf{x} - \boldsymbol{\nu}_\ell)\right)^{\eta_\ell}\right). \quad (27)$$

The moments and a more standard parametrization are discussed in Appendix A.4. Generalized Gaussian mixture pdfs are popular in Bayesian inference approaches for image recovery [69, 70, 71, 72], as they constitute a rich and accurate model for texture components as well as for wavelet coefficient distributions. In this experiment, we set  $L = 5$  components, with means  $\boldsymbol{\gamma}_1 = [-10, -10]^\top$ ,  $\boldsymbol{\gamma}_2 = [0, 16]^\top$ ,  $\boldsymbol{\gamma}_3 = [13, 8]^\top$ ,  $\boldsymbol{\gamma}_4 = [-9, 7]^\top$ ,  $\boldsymbol{\gamma}_5 = [14, -4]^\top$ ,  $\boldsymbol{\Sigma}_\ell = I_{d_x}$ ,  $\omega_\ell = 1/5$ , and  $\eta_\ell = \eta \in \{0.5, 1, 1.5\}$ , for all  $\ell = 1, \dots, 5$ . The location parameters of the proposals are now randomly initialized in the square  $[13, 15] \times [-6, -8]$ . This initialization is particularly adversarial, since the location parameters of the proposals are very concentrated around only one of the five modes. We use the same parameters as in the previous example, except that GRAMIS now implements  $G_t = 1$  to enforce exploration via the repulsion term. We compare GRAMIS with AMIS [10], LR-PMC [20], and O-PMC [23] algorithms. We set the same parameters in all algorithms except in AMIS where, since it has only one proposal, we set  $K = 1000$  so all methods have the same number of target evaluations. Note that the generalized Gaussian pdf (27) is not differentiable at its mean as soon as  $\nu < 1$  (e.g.,  $\nu = 0.5$  is Laplace distribution). In Appendix A.1, we discuss how to circumvent this issue by building a smoothed approximation of the target, and explicit its gradients and Hessians (in the experiment we set  $\delta = 10^{-5}$  for this smoothed version). Note that, for  $\nu < 0.5$ , the Hessian is not necessarily definite positive in all  $\mathbf{x} \in \mathbb{R}^{d_x}$ . However, this does not create any instability in our implementation, thanks to the safe rule defined in (13) for the covariance adaptation.

The results are displayed in Table 4 in terms of RMSE in the estimation of  $Z$  and the first and second moments. We can see that GRAMIS outperforms all competitors in all targets except in the second moment for the case with  $\eta = 0.5$  (Laplace distributions), where LR-PMC, O-PMC, and GRAMIS perform very similarly. This target has heavier tails and allows all algorithms to better explore the space compared to the cases with  $\eta = 1$  (Gaussian distributions) and  $\eta = 1.5$  (lighter tails). In this two last scenarios, all

competitors get stuck in one or two modes while the repulsion of GRAMIS allows it to still discover the five modes. Finally, Table 4 also includes the estimated  $\chi^2(\tilde{\pi}, \psi^{(T)})$  divergence, where  $\psi^{(T)}(\mathbf{x}) = \frac{1}{N} \sum_{n=1}^N q_n^{(T)}(\mathbf{x})$  denotes the equally-weighted mixture of proposals in the last iteration. We averaged the estimated divergence over all the independent runs. We note also that the  $\chi^2$  divergence is of particular interest in the case of importance sampling, since the variance of the estimator of the normalizing constant can be expressed as  $\text{Var}(\hat{Z}) = \frac{Z^2}{N} \chi^2(\tilde{\pi}, \psi^{(T)})$  [73, 74, 75]. Thus, it is natural that in all cases, the ranking in performance is the same as for the RMSE of the estimator  $\hat{Z}$ .

RMSE	AMIS			LR-PMC			O-PMC			GRAMIS		
	$\eta = 0.5$	$\eta = 1$	$\eta = 1.5$	$\eta = 0.5$	$\eta = 1$	$\eta = 1.5$	$\eta = 0.5$	$\eta = 1$	$\eta = 1.5$	$\eta = 0.5$	$\eta = 1$	$\eta = 1.5$
Z	0.9692	0.9692	0.9693	0.0123	0.5145	0.6103	0.0091	0.6400	0.6261	<b>6.43·10<sup>-4</sup></b>	<b>2.46·10<sup>-6</sup></b>	<b>2.57·10<sup>-3</sup></b>
$E_{\tilde{\pi}}[\mathbf{X}]$	56.08	56.07	56.13	1.2345	50.49	55.17	0.6225	56.09	55.51	<b>0.0249</b>	<b>1.49·10<sup>-4</sup></b>	<b>0.1097</b>
$E_{\tilde{\pi}}[\mathbf{X}^2]$	67.95	67.93	68.03	1.119	55.57	65.80	<b>1.0285</b>	67.89	66.62	1.2957	<b>0.0020</b>	<b>0.1875</b>
$\chi^2(\tilde{\pi}, \psi^{(T)})$	980.12	980.07	980.08	30.61	556.74	616.8	7.836	639.93	630.27	<b>0.7781</b>	<b>0.0053</b>	<b>4.3387</b>

Table 4: **Mixture of generalized Gaussians in Section 4.2.** RMSE of the IS estimators. In the case of GRAMIS,  $G_1 = 1$  with exponential decay. The MSE results are obtained over 100 independent runs, with estimators using the weighted samples on the half last iterations.

### 4.3. Banana-shaped distribution

We now consider a banana-shaped distribution [76, 77]. The shape of this target makes it particularly challenging for sampling methods. The target is the pdf of a r.v. resulting from a transformation of a  $d_x$ -dimensional multivariate Gaussian r.v.  $\bar{\mathbf{X}} \sim \mathcal{N}(\mathbf{x}; \mathbf{0}_{d_x}, \Sigma)$  with  $\Sigma = \text{diag}(c^2, 1, \dots, 1)$ . The transformed r.v. is  $(X_j)_{1 \leq j \leq d_x}$  such that  $X_j = \bar{X}_j$  for  $j \in \{1, \dots, d_x\} \setminus 2$ , and  $X_2 = \bar{X}_2 - b(\bar{X}_1^2 - c^2)$ , where we set  $c = 1$  and  $b = 3$  in our example.

First, we consider a toy example with  $d_x = 2$  so we can obtain intuitive plots. We set  $T = 100$ ,  $N = 50$ , and  $K = 20$ . Figure 2 shows the the target distribution, the final location parameters (black dots) and scale parameters (black ellipses) of the proposals at time  $T = 100$ , and the samples of the last iteration (red dots). We consider the GRAMIS scheme with constant repulsion, with  $G_t \in \{0.02, 0.01, 0.005, 0.001, 0.0001, 0\}$ . It can be seen that bigger values of  $G_t$  yield effectively a mixture with well separated

proposals. In this example, the proposals remain in practice static after a few iterations. When  $G_t$  is smaller, the proposals tend to concentrate around the mode. In the extreme case with  $G_t = 0$  (i.e., without) repulsion, all proposals are effectively the same, which in practice coincides with the Laplace approximation [78].

We now perform comparisons with competitive algorithms, namely PMC using either global (GR) or local (LR) resampling [20], AMIS [10], and O-PMC with LR [23], for various dimension  $d_x$ . In AMIS, we set  $N = 1$ ,  $K = 500$  and  $T = 40$ . The other algorithms set  $N = 50$ ,  $K = 20$ , and  $T = 20$ , so all algorithms have the same number of target evaluations. We measure the MSE of all algorithms in estimating  $E_{\tilde{\pi}}[\mathbf{X}]$ . In all cases, we select Gaussian proposal densities, with location parameters that are randomly initialized within the square  $[-4, 4] \times [-4, 4]$ . All algorithms are initialized with isotropic proposal covariances with  $\sigma \in \{1, 3, 5\}$ .

Table 5 shows the MSE of the proposed GRAMIS in the estimation of the target mean for  $d_x \in \{5, 20, 50\}$ . In Fig. 3, we compare GRAMIS, with LR-PMC, GR-PMC and O-PMC in a range of dimensions  $d_x \in \{2, 5, 10, 15, 20, 30, 40, 50\}$ . The best performance is reached by GRAMIS in all dimensions, followed by the LR version of the O-PMC. We implement in this example a simple version of GRAMIS without repulsion, which simplifies the parameter tuning for different dimensions. In our GRAMIS and in O-PMC, the MSE tends to decrease when the dimensions grows, which can be explained by the strong structure of the banana-shaped target in high dimensions.

Finally, Fig. (4) shows the MSE in the estimation of  $E_{\tilde{\pi}}[\mathbf{X}]$  versus the number of total iterations  $T \in [10, 200]$  for the proposed GRAMIS method (with  $\sigma = 1$ ) using setting values of  $G_{\text{rep}} \in \{0, 10^{-2}\}$ . The estimators are built in all cases by using the last half of the total iterations. In the standard GRAMIS with  $G_{\text{rep}} = 10^{-2}$ , increasing the number of iterations improves the adaptation and thus the performance. However, if  $G_{\text{rep}} = 0$  (i.e., no repulsion), running the algorithm for more iterations worsens the performance. This can be understood by seeing the last plot in Fig. (2). In this case, all proposals tend to be the same, and thus the mixture does not represent well the whole target.

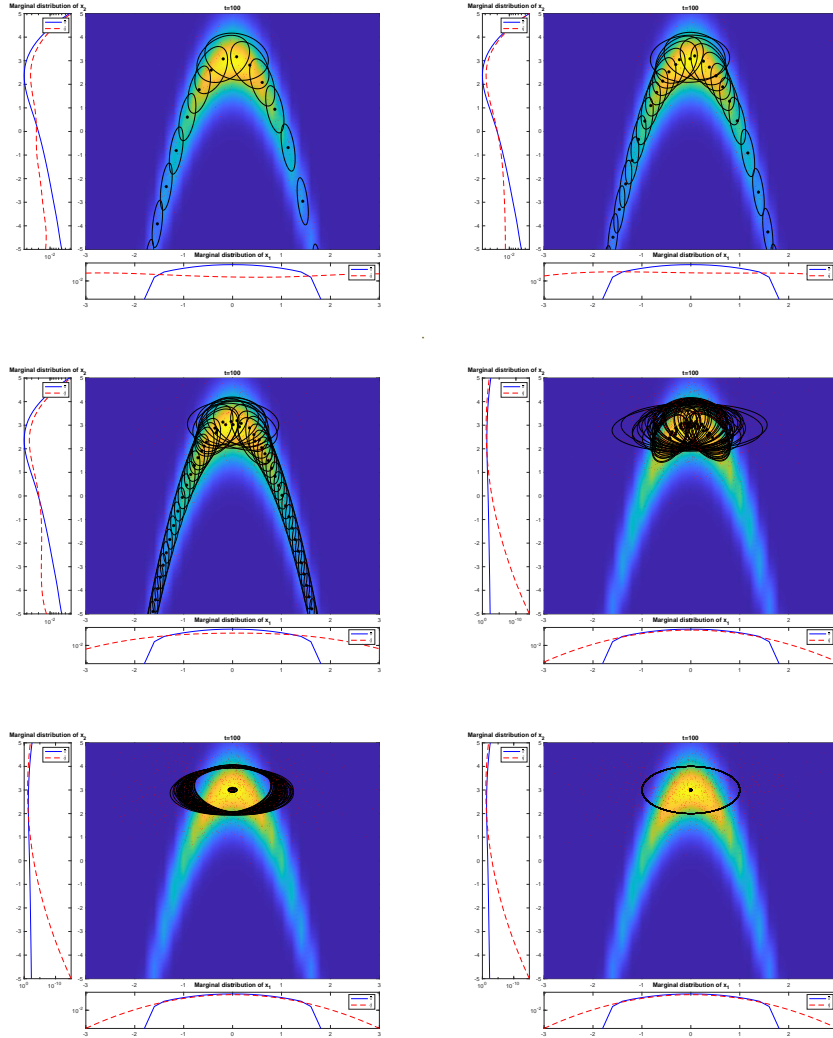


Figure 2: **Banana-shaped target in Section 4.3.** Final location parameters (black dots) and scale parameters (black ellipses) of the proposals at time  $T = 100$  for six ablated versions GRAMIS with constant repulsion. In order:  $G_T \in \{0.02, 0.01, 0.005, 0.001, 0.0001, 0\}$ .

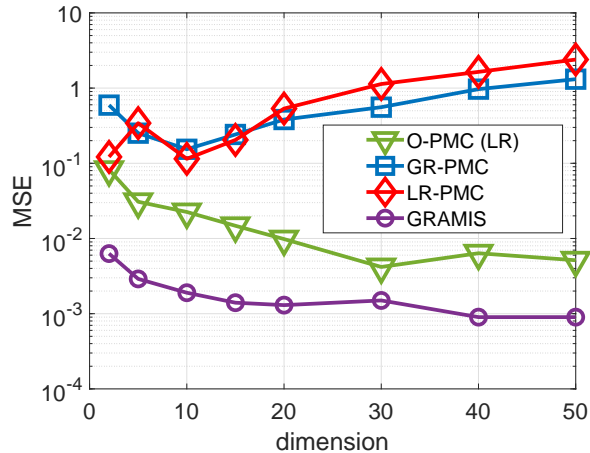


Figure 3: **Banana-shaped target in Section 4.3.** MSE in the estimation of  $E_{\tilde{\pi}}[\mathbf{X}]$  versus the dimension  $d_x$ , with GR-PMC, LR-PMC, O-PMC (using LR), and the proposed GRAMIS method (with  $\sigma = 1$ ).

	GR-PMC	LR-PMC	AMIS	LR-O-PMC	GAPIS	GRAMIS
$d_x = 5$	0.2515	0.3418	0.1758	0.0308	0.3007	<b>0.0029</b>
$d_x = 20$	0.3818	0.5340	0.1901	0.0098	1.5299	<b>0.0013</b>
$d_x = 50$	1.3134	2.3963	0.6074	0.0051	2.5524	<b>0.0009</b>

Table 5: **Banana-shaped target in Section 4.3.** MSE in the estimation of  $E_{\tilde{\pi}}[\mathbf{X}]$  of the banana-shaped distribution for dimensions  $d_x = 5, 20$  and  $50$ . For all methods, we set the initial proposal variance to  $\sigma = 1$ . In all PMC-based methods,  $(N, K, T) = (50, 20, 20)$  while  $(N, K, T) = (1, 500, 40)$  for AMIS.

## 5. Conclusion

In this paper, we have proposed a new algorithm, called GRAMIS, that iteratively adapts a set of proposals with the goal of improving the performance of the importance sampling estimators. The geometric information of the target is exploited by using the first-order and second-order information to improve the location and the scale parameters. A cooperation in the adaptation is allowed by introducing a repulsion term, which can be justified through the lens of Poisson fields. This repulsion becomes essential in multi-modal scenarios and also to represent target densities that, even if uni-modal, cannot be well approximated by standard uni-modal proposals. The GRAMIS algorithm exhibits good exploratory capabilities and a powerful representation of compli-



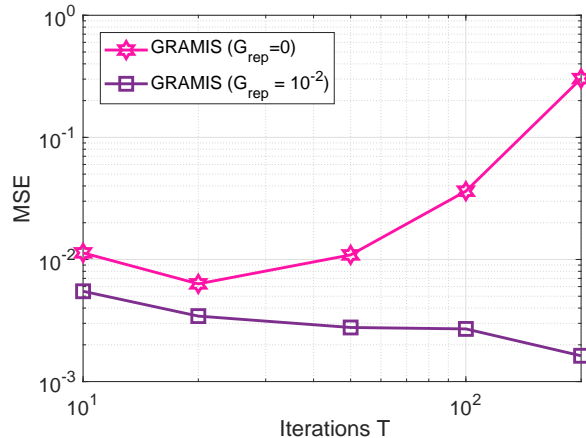


Figure 4: **Banana-shaped target in Section 4.3.** MSE in the estimation of  $E_{\tilde{\pi}}[\mathbf{X}]$  versus the number of iterations  $T$  for the proposed GRAMIS method (with  $\sigma = 1$ ) setting  $G_{\text{rep}} \in \{0, 10^{-2}\}$ .

cated target densities, leading in most cases to lower-variance estimators compared to other AIS algorithms. As a future work, we plan to analyze the behaviour of the repulsion term in the proposals in GRAMIS algorithm in connection with a discretised Poisson field.

### Acknowledgement

The work of V. E. is supported by the *Agence Nationale de la Recherche* of France under PISCES (ANR-17-CE40-0031-01), the Leverhulme Research Fellowship (RF-2021-593), and by ARL/ARO under grants W911NF-20-1-0126 and W911NF-22-1-0235.

## Appendix A. Gradient, Hessian, moments, and re-parametrization for the generalized Gaussian distribution

### Appendix A.1. Smoothed approximation

We define the smoothed version of the multivariate generalized Gaussian distribution, as introduced in [79] under the name *generalized multivariate exponential power*

prior (GMEP):

$$(\forall \mathbf{x} \in \mathbb{R}^{d_x}) \quad \mathcal{GM}\mathcal{EP}(\mathbf{x}; \boldsymbol{\nu}, \boldsymbol{\Sigma}, \boldsymbol{\eta}, \delta) = C |\boldsymbol{\Sigma}|^{-1/2} \exp\left(-\frac{1}{2} \left( (\mathbf{x} - \boldsymbol{\nu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\nu}) + \delta \right)^\eta\right), \quad (\text{A.1})$$

with  $\delta > 0$  and  $C$  an appropriate normalization constant. The new parameter  $\delta$  allows to smooth the distribution at  $\mathbf{x} = \boldsymbol{\nu}$ , so that (A.1) is now twice differentiable on  $\mathbb{R}^{d_x}$ . The original generalized Gaussian pdf is the case with  $\delta = 0$ . Note that the smoothing preserves the elliptical shape of the distribution. The practical application of such distribution to Bayesian inference for image recovery and remote sensing has been illustrated, for instance, in [80, 36, 81].

#### Appendix A.2. Gradient expression

We aim at calculating the gradient of  $\log \tilde{\pi}$ , with  $\tilde{\pi}$  defined as in (26). As aforementioned, we make use of the smoothed version (A.1), so that

$$(\forall \mathbf{x} \in \mathbb{R}^{d_x}) \quad \nabla(\log \tilde{\pi})(\mathbf{x}) = \frac{1}{\tilde{\pi}(\mathbf{x})} \nabla \tilde{\pi}(\mathbf{x}), \quad (\text{A.2})$$

with

$$(\forall \mathbf{x} \in \mathbb{R}^{d_x}) \quad \nabla \tilde{\pi}(\mathbf{x}) = \sum_{\ell=1}^L \omega_\ell \nabla \mathcal{GG}(\mathbf{x}; \boldsymbol{\nu}_\ell, \boldsymbol{\Sigma}_\ell, \boldsymbol{\eta}_\ell) \quad (\text{A.3})$$

$$\approx \sum_{\ell=1}^L \omega_\ell \nabla \mathcal{GM}\mathcal{EP}(\mathbf{x}; \boldsymbol{\nu}_\ell, \boldsymbol{\Sigma}_\ell, \boldsymbol{\eta}_\ell, \delta). \quad (\text{A.4})$$

Let us now explicit the gradient of (A.1). We introduce the short notation

$$(\forall \mathbf{x} \in \mathbb{R}^{d_x})(\forall \ell \in \{1, \dots, L\}) \quad \boldsymbol{\theta}_\ell(\mathbf{x}) = (\mathbf{x} - \boldsymbol{\nu}_\ell)^\top \boldsymbol{\Sigma}_\ell^{-1} (\mathbf{x} - \boldsymbol{\nu}_\ell), \quad (\text{A.5})$$

and

$$(\forall y \in \mathbb{R})(\forall \ell \in \{1, \dots, L\}) \quad g_\ell(y) = C_\ell \exp\left(-\frac{(y + \delta)^\eta}{2}\right), \quad (\text{A.6})$$

so that

$$(\forall \mathbf{x} \in \mathbb{R}^{d_x}) \quad \mathcal{GM}\mathcal{EP}(\mathbf{x}; \boldsymbol{\nu}, \boldsymbol{\Sigma}, \boldsymbol{\eta}, \delta) = (g_\ell \circ \boldsymbol{\theta}_\ell)(\mathbf{x}). \quad (\text{A.7})$$

Finally, using the chain rule:

$$(\forall \mathbf{x} \in \mathbb{R}^{d_x})(\forall \ell \in \{1, \dots, L\}) \quad \nabla(g_\ell \circ \boldsymbol{\theta}_\ell)(\mathbf{x}) = g'_\ell(\boldsymbol{\theta}_\ell(\mathbf{x})) \times \nabla \boldsymbol{\theta}_\ell(\mathbf{x}), \quad (\text{A.8})$$

with the first derivative expression

$$(\forall y \in \mathbb{R})(\forall \ell \in \{1, \dots, L\}) \quad g'_\ell(y) = -\frac{\eta_\ell}{2}(y + \delta)^{\eta_\ell - 1} g_\ell(y), \quad (\text{A.9})$$

and the gradient

$$(\forall \mathbf{x} \in \mathbb{R}^{d_x})(\forall \ell \in \{1, \dots, L\}) \quad \nabla \theta_\ell(\mathbf{x}) = 2\Sigma_\ell^{-1}(\mathbf{x} - \nu_\ell). \quad (\text{A.10})$$

### Appendix A.3. Hessian expression

We proceed in a similar manner, using the smoothed version of the generalised Gaussian distribution. First,

$$(\forall \mathbf{x} \in \mathbb{R}^{d_x}) \quad \nabla^2(\log \tilde{\pi})(\mathbf{x}) = -\frac{1}{\tilde{\pi}(\mathbf{x})^2} \nabla \tilde{\pi}(\mathbf{x}) \nabla \tilde{\pi}(\mathbf{x})^\top + \frac{1}{\tilde{\pi}(\mathbf{x})} \nabla^2 \tilde{\pi}(\mathbf{x}), \quad (\text{A.11})$$

with

$$(\forall \mathbf{x} \in \mathbb{R}^{d_x}) \quad \nabla^2 \tilde{\pi}(\mathbf{x}) = \sum_{\ell=1}^L \omega_\ell \nabla^2 \mathcal{G}(\mathbf{x}; \nu_\ell, \Sigma_\ell, \eta_\ell), \quad (\text{A.12})$$

$$\approx \sum_{\ell=1}^L \omega_\ell \nabla^2 \mathcal{GMEP}(\mathbf{x}; \nu_\ell, \Sigma_\ell, \eta_\ell, \delta). \quad (\text{A.13})$$

Again, using the chain rule we obtain

$$\begin{aligned} (\forall \mathbf{x} \in \mathbb{R}^{d_x})(\forall \ell \in \{1, \dots, L\}) \quad \nabla^2(g_\ell \circ \theta_\ell)(\mathbf{x}) &= g''_\ell(\theta_\ell(\mathbf{x})) \times \nabla \theta_\ell(\mathbf{x}) \nabla \theta_\ell(\mathbf{x})^\top \\ &\quad + g'_\ell(\theta_\ell(\mathbf{x})) \times \nabla^2 \theta_\ell(\mathbf{x}), \end{aligned} \quad (\text{A.14})$$

where we have used the expressions

$$(\forall \mathbf{x} \in \mathbb{R}^{d_x})(\forall \ell \in \{1, \dots, L\}) \quad \nabla^2 \theta_\ell(\mathbf{x}) = 2\Sigma_\ell^{-1}, \quad (\text{A.15})$$

and the second order derivative

$$(\forall y \in \mathbb{R})(\forall \ell \in \{1, \dots, L\}) \quad g''_\ell(y) = \left( -\frac{\eta_\ell(\eta_\ell - 1)}{2}(y + \delta)^{\eta_\ell - 2} + \frac{\eta_\ell^2}{4}(y + \delta)^{2\eta_\ell - 2} \right) g_\ell(y). \quad (\text{A.16})$$

### Appendix A.4. Moments and re-parametrization

The generalized Gaussian distribution parametrized in (27) has a mean  $E[X] = \nu_\ell$  and a covariance  $\text{Cov}[X] = \frac{2^{\frac{1}{\eta_\ell}} \Gamma\left(\frac{d_x + 2}{2\eta_\ell}\right)}{d_x \Gamma\left(\frac{d_x}{2\eta_\ell}\right)} \Sigma_\ell$ .

A common parametrization for the scalar generalized Gaussian distribution is given by  $\mathcal{GG}(x; \nu, \alpha, \beta) = \frac{\beta}{2\alpha\Gamma(1/\beta)} e^{-(|x-\mu|/\alpha)^\beta}$  instead of our choice  $\mathcal{GG}(x; \nu, \sigma, \eta) = \frac{1}{\Gamma(\frac{1}{2\eta})2^{\frac{1}{2\eta}}\sigma} e^{-\frac{1}{2}\left(\frac{|x-\nu|}{\sigma}\right)^{2\eta}}$ . It is easy to see by identification that the re-parametrization corresponds to  $\beta = 2\eta$  and  $\alpha = 2^{\frac{1}{2\eta}}\sigma$ .

## References

- [1] C. P. Robert, G. Casella, Monte Carlo Statistical Methods, Springer-Verlag New York, 2004.
- [2] J. S. Liu, Monte Carlo Strategies in Scientific Computing, Springer-Verlag New York, 2004.
- [3] A. Owen, Monte Carlo Theory, Methods and Examples, <http://statweb.stanford.edu/~owen/mc/>, 2013.
- [4] V. Elvira, L. Martino, Advances in importance sampling, Wiley StatsRef: Statistics Reference Online (2021) 1–22.
- [5] V. Elvira, L. Martino, D. Luengo, M. F. Bugallo, Generalized multiple importance sampling, *Statistical Science* 34 (1) (2019) 129–155.
- [6] M. F. Bugallo, V. Elvira, L. Martino, D. Luengo, J. Míguez, P. M. Djuric, Adaptive importance sampling: The past, the present, and the future, *IEEE Signal Processing Magazine* 34 (4) (2017) 60–79.
- [7] R. Douc, A. Guillin, J. M. Marin, C. P. Robert, Convergence of adaptive mixtures of importance sampling schemes, *Annals of Statistics* 35 (2007) 420–448.
- [8] O. D. Akyildiz, J. Míguez, Convergence rates for optimised adaptive importance samplers, *Statistics and Computing* 31 (2) (2021) 1–17.
- [9] Ö. D. Akyildiz, Global convergence of optimized adaptive importance samplers, arXiv preprint arXiv:2201.00409 (2022).
- [10] J. M. Cornuet, J. M. Marin, A. Mira, C. P. Robert, Adaptive multiple importance sampling, *Scandinavian Journal of Statistics* 39 (4) (2012) 798–812.

- [11] J.-M. Marin, P. Pudlo, M. Sedki, Consistency of adaptive importance sampling and recycling schemes, *Bernoulli* 25 (3) (2019) 1977–1998.
- [12] L. Martino, V. Elvira, D. Luengo, J. Corander, An adaptive population importance sampler, *IEEE International Conf. on Acoustics, Speech, and Signal Processing (ICASSP)* (2014) 8088–8092.
- [13] T. Paananen, J. Piironen, P.-C. Bürkner, A. Vehtari, Implicitly adaptive importance sampling, *Statistics and Computing* 31 (2) (2021) 1–19.
- [14] R. Douc, O. Cappé, E. Moulines, Comparison of resampling schemes for particle filtering, in: *Proceedings of the 4<sup>th</sup> International Symposium on Image and Signal Processing and Analysis (ISPA 2005)*, Zagreb, Croatia, 2005, pp. 64–69.
- [15] T. Li, M. Bolic, P. M. Djuric, Resampling methods for particle filtering: Classification, implementation, and strategies, *IEEE Signal Processing Magazine* 32 (3) (2015) 70–86.
- [16] O. Cappé, A. Guillin, J. M. Marin, C. P. Robert, Population Monte Carlo, *Journal of Computational and Graphical Statistics* 13 (4) (2004) 907–929.
- [17] O. Cappé, R. Douc, A. Guillin, J. M. Marin, C. P. Robert, Adaptive importance sampling in general mixture classes, *Statistical Computing* 18 (2008) 447–459.
- [18] E. Koblents, J. Miguez, Robust mixture population Monte Carlo scheme with adaptation of the number of components, in: *Proceedings of the 21st European Signal Processing Conference (EUSIPCO 2013)*, Marrakech, Morocco, 2013, pp. 1–5.
- [19] L. Martino, V. Elvira, J. Míguez, A. Artés-Rodríguez, P. Djurić, A comparison of clipping strategies for importance sampling, in: *2018 IEEE Statistical Signal Processing Workshop (SSP)*, IEEE, 2018, pp. 558–562.
- [20] V. Elvira, L. Martino, D. Luengo, M. F. Bugallo, Improving Population Monte Carlo: Alternative weighting and resampling schemes, *Signal Processing* 131 (12) (2017) 77–91.

- [21] V. Elvira, L. Martino, D. Luengo, 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), IEEE, 2017, pp. 1–5.
- [22] C. Miller, J. N. Corcoran, M. D. Schneider, Rare events via cross-entropy population monte carlo, *IEEE Signal Processing Letters* 29 (2021) 439–443.
- [23] V. Elvira, E. Chouzenoux, Optimized population monte carlo, *IEEE Transactions on Signal Processing* 70 (2022) 2489–2501.
- [24] L. Martino, V. Elvira, D. Luengo, J. Corander, Layered adaptive importance sampling, *Statistics and Computing* 27 (3) (2015) 599–623.
- [25] I. Schuster, I. Klebanov, Markov chain importance sampling - a highly efficient estimator for MCMC, (to appear) *Journal of Computational and Graphical Statistics* <https://arxiv.org/abs/1805.07179> (2021).
- [26] D. Rudolf, B. Sprungk, On a Metropolis–Hastings importance sampling estimator, *Electronic Journal of Statistics* 14 (1) (2020) 857–889.
- [27] A. Mousavi, R. Monsefi, V. Elvira, Hamiltonian adaptive importance sampling, *IEEE Signal Processing Letters* 28 (2021) 713–717.
- [28] M. Pereyra, P. Schniter, E. Chouzenoux, J.-C. Pesquet, J.-Y. Tournier, A. Hero, S. McLaughlin, A survey of stochastic simulation and optimization methods in signal processing, *IEEE Journal on Selected Topics in Signal Processing* 10 (2) (2016) 224–241.
- [29] G. O. Roberts, O. Stramer, Langevin diffusions and Metropolis-Hastings algorithms, *Methodology and Computing in Applied Probability* 4 (4) (2002) 337–357.
- [30] A. Durmus, E. Moulines, M. Pereyra, Efficient Bayesian computation by proximal Markov chain Monte Carlo: when Langevin meets Moreau, *SIAM Journal on Imaging Sciences* 11 (1) (2018) 473–506.

- [31] A. Schreck, G. Fort, S. L. Corff, E. Moulines, A shrinkage-thresholding Metropolis adjusted Langevin algorithm for Bayesian variable selection, *IEEE Journal on Selected Topics in Signal Processing* 10 (2) (2016) 366–375.
- [32] M. Girolami, B. Calderhead, Riemann manifold Langevin and Hamiltonian Monte Carlo methods, *Journal of the Royal Statistical Society Series B (Statistical Methodology)* 73 (91) (2011) 123–214.
- [33] J. Martin, C. L. Wilcox, C. Burstedde, O. Ghattas, A stochastic Newton MCMC method for large-scale statistical inverse problems with application to seismic inversion, *SIAM Journal on Scientific Computing* 34 (3) (2012) 1460–1487.
- [34] Y. Zhang, C. A. Sutton, Quasi-Newton methods for Markov chain Monte Carlo, in: *Proceedings of the Neural Information Processing Systems workshop (NIPS 2011)*, no. 24, Granada, Spain, 2011, pp. 2393–2401.
- [35] Y. Qi, T. P. Minka, Hessian-based Markov Chain Monte-Carlo algorithms, *Proceedings of the First Cape Cod Workshop on Monte Carlo Methods* (2002).
- [36] Y. Marnissi, E. Chouzenoux, A. Benazza-Benyahia, J.-C. Pesquet, Majorize-Minimize adapted Metropolis-Hastings algorithm, *IEEE Transactions on Signal Processing* (68) (2020) 2356–2369.
- [37] Y. Marnissi, A. Benazza-Benyahia, E. Chouzenoux, J.-C. Pesquet, Majorize-Minimize adapted Metropolis Hastings algorithm. application to multichannel image recovery, in: *Proceedings of the 22nd European Signal Processing Conference (EUSIPCO 2014)*, Lisboa, Portugal, 2014, pp. 1332–1336.
- [38] Y. Marnissi, E. Chouzenoux, A. Benazza-Benyahia, J.-C. Pesquet, An auxiliary variable method for MCMC algorithms in high dimension, *Entropy* 20 (110) (2018).
- [39] U. Simsekli, R. Badeau, A. T. Cemgil, G. Richard, Stochastic quasi-Newton Langevin Monte Carlo, in: *Proceedings of the 33rd International Conference on International Conference on Machine Learning (ICML 2016)*, Vol. 48, 2016, p. 642–651.

- [40] I. Schuster, Gradient importance sampling, Tech. rep., <https://arxiv.org/abs/1507.05781> (2015).
- [41] M. Fasiolo, F. E. de Melo, S. Maskell, Langevin incremental mixture importance sampling, *Statistical Computing* 28 (3) (2018) 549–561.
- [42] G. O. Roberts, L. R. Tweedie, Exponential convergence of Langevin distributions and their discrete approximations, *Bernoulli* 2 (4) (1996) 341–363.
- [43] V. Gallego, D. R. Insua, Stochastic gradient mcmc with repulsive forces, arXiv preprint arXiv:1812.00071 (2018).
- [44] Y. El-Laham, V. Elvira, M. F. Bugallo, Robust covariance adaptation in adaptive importance sampling, *IEEE Signal Processing Letters* (2018).
- [45] Y. El-Laham, V. Elvira, M. Bugallo, Recursive shrinkage covariance learning in adaptive importance sampling, in: *Proceedings of the 8th IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP 2019)*, Guadeloupe, France, 2019, pp. 624–628.
- [46] E. Veach, L. Guibas, Optimally combining sampling techniques for Monte Carlo rendering, in: *Proceedings of the 22nd International ACM Conference on Computer Graphics and Interactive Techniques (SIGGRAPH 1995)*, 1995, pp. 419–428.
- [47] A. Owen, Y. Zhou, Safe and effective importance sampling, *Journal of the American Statistical Association* 95 (449) (2000) 135–143.
- [48] M. Sbert, V. Elvira, Generalizing the balance heuristic estimator in multiple importance sampling, *Entropy* 24 (2) (2022) 191.
- [49] L. Martino, V. Elvira, D. Luengo, J. Corander, An adaptive population importance sampler: Learning from the uncertainty, *IEEE Transactions on Signal Processing* 63 (16) (2015) 4422–4437.
- [50] V. Elvira, L. Martino, L. Luengo, J. Corander, A gradient adaptive population importance sampler, in: *Proceedings of the IEEE International Conference on*



- Acoustics, Speech and Signal Processing (ICASSP 2015), Brisbane, Australia, 2015, pp. 4075–4079.
- [51] M. Welling, Y. W. Teh, Bayesian learning via stochastic gradient langevin dynamics, in: Proceedings of the 28th International Conference on Machine Learning (ICML 2011), 2011.
- [52] C. Rasmussen, A practical Monte Carlo implementation of bayesian learning, in: Proceedings of the Neural Information Processing Systems Conference (NIPS 1995), 1995.
- [53] T. Hesterberg, Weighted average importance sampling and defensive mixture distributions, *Technometrics* 37 (2) (1995) 185–194.
- [54] I. Kondapaneni, P. Vévoda, P. Grittmann, T. Skřivan, P. Slusallek, J. Křivánek, Optimal multiple importance sampling, *ACM Transactions on Graphics (TOG)* 38 (4) (2019) 1–14.
- [55] M. Sbert, V. Havran, L. Szirmay-Kalos, Multiple importance sampling revisited: breaking the bounds, *EURASIP Journal on Advances in Signal Processing* 2018 (1) (2018) 1–15.
- [56] V. Elvira, L. Martino, D. Luengo, M. F. Bugallo, Efficient multiple importance sampling estimators, *Signal Processing Letters, IEEE* 22 (10) (2015) 1757–1761.
- [57] V. Elvira, L. Martino, D. Luengo, M. F. Bugallo, Heretical multiple importance sampling, *IEEE Signal Processing Letters* 23 (10) (2016) 1474–1478.
- [58] V. Elvira, L. Martino, D. Luengo, M. F. Bugallo, Multiple importance sampling with overlapping sets of proposals, in: 2016 IEEE Statistical Signal Processing Workshop (SSP), IEEE, 2016, pp. 1–5.
- [59] D. Talay, L. Tubaro, Expansion of the global error for numerical schemes solving stochastic differential equations, *Stochastic Analysis and Applications* 8 (4) (1991) 483–509.

- [60] A. Durmus, E. Moulines, High-dimensional Bayesian inference via the unadjusted Langevin algorithm, *Bernoulli* (4A) (2019) 2854–2882.
- [61] C. Vacar, J.-F. Giovannelli, Y. Berthoumieu, Langevin and Hessian with Fisher approximation stochastic sampling for parameter estimation of structured covariance, in: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2011)*, Prague, Czech Republic, 2011, pp. 3964–3967.
- [62] T. Xifara, C. Sherlock, S. Livingstone, S. Byrne, M. Girolami, Langevin diffusions and the Metropolis-adjusted Langevin algorithm, *Statistics and Probability Letters* 91 (2014) 14 – 19.
- [63] S. Sabanis, Y. Zhang, Higher order Langevin Monte Carlo algorithm, *Electronic Journal of Statistics* 13 (2) (2019) 3805–3850.
- [64] L. Martino, V. Elvira, D. Luengo, A. Artes-Rodriguez, J. Corander, Orthogonal MCMC algorithms, in: *2014 IEEE Workshop on Statistical Signal Processing (SSP)*, IEEE, 2014, pp. 364–367.
- [65] L. Martino, V. Elvira, D. Luengo, A. Artés-Rodríguez, J. Corander, Smelly parallel mcmc chains, in: *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2015, pp. 4070–4074.
- [66] J. Nocedal, S. Wright, *Numerical Optimization*, 2nd Edition, Springer, New York, NY, 2006.
- [67] Y. Xu, Z. Liu, M. Tegmark, T. S. Jaakkola, Poisson flow generative models, in: *Advances in Neural Information Processing Systems*, 2022.
- [68] E. Gomez, M. Gomez-Viilegas, J. Marn, A multivariate generalization of the power exponential family of distributions, *Communications in Statistics - Theory and Methods* 27 (3) (1998) 589–600.
- [69] C.-A. Deledalle, S. Parameswaran, T. Q. Nguyen, Image denoising with generalized gaussian mixture model patch priors, *SIAM Journal on Imaging Sciences* 11 (4) (2018) 2568–2609.

- [70] S.-K. S. Fan, Y. Lin, A fast estimation method for the generalized gaussian mixture distribution on complex images, *Computer Vision and Image Understanding* 113 (7) (2009) 839–853.
- [71] D.-P.-L. Nguyen, J.-F. Aujol, Y. Berthoumieu, Patch-based image super resolution using generalized gaussian mixture model, *arXiv preprint arXiv:2206.03069* (2022).
- [72] M.-C. Corbineau, D. Kouamé, E. Chouzenoux, J.-Y. Tourneret, J.-C. Pesquet, Preconditioned p-ula for joint deconvolution-segmentation of ultrasound images, *IEEE Signal Processing Letters* 10 (26) (2019) 1456–1460.
- [73] E. K. Ryu, S. P. Boyd, Adaptive importance sampling via stochastic convex programming, *arXiv preprint arXiv:1412.4845* (2014).
- [74] S. Agapiou, O. Papaspiliopoulos, D. Sanz-Alonso, A. Stuart, et al., Importance sampling: Intrinsic dimension and computational cost, *Statistical Science* 32 (3) (2017) 405–431.
- [75] J. Míguez, On the performance of nonlinear importance samplers and population Monte Carlo schemes, in: *2017 22nd International Conference on Digital Signal Processing (DSP)*, IEEE, 2017, pp. 1–5.
- [76] H. Haario, E. Saksman, J. Tamminen, Adaptive proposal distribution for random walk Metropolis algorithm, *Computational Statistics* 14 (3) (1999) 375–396.
- [77] H. Haario, E. Saksman, J. Tamminen, An adaptive Metropolis algorithm, *Bernoulli* 7 (2) (2001) 223–242.
- [78] Z. Shun, P. McCullagh, Laplace approximation of high dimensional integrals, *Journal of the Royal Statistical Society: Series B (Methodological)* 57 (4) (1995) 749–760.
- [79] Y. Marnissi, A. Benazza-Benyahia, E. Chouzenoux, J.-C. Pesquet, Generalized multivariate exponential power prior for wavelet-based multichannel image

restoration, in: Proceedings of the 20th IEEE International Conference on Image Processing (ICIP 2013), Melbourne, Australia, 2013, pp. 2402–2406.

- [80] Y. Marnissi, A. Benazza-Benyahia, E. Chouzenoux, J.-C. Pesquet, Majorize-minimize adapted metropolis hastings algorithm. application to multichannel image recovery, in: Proceedings of the 22nd European Signal Processing Conference (EUSIPCO 2014), Lisboa, Portugal, 2014, pp. 1332–1336.
- [81] Y. Marnissi, E. Chouzenoux, A. Benazza-Benyahia, J.-C. Pesquet, An auxiliary variable method for MCMC algorithms in high dimension, *Entropy* 20 (110) (2018).