Efficient Adaptive Multiple Importance Sampling

Efficient Adaptive Multiple Importance Sampling

Yousef El-Laham[◇], Luca Martino[□], Víctor Elvira[⋆], Mónica F. Bugallo[◇]
[◇] Dept. of Electrical and Computer Engineering, Stony Brook University, New York (USA)
[⋆] IMT Lille Douai, CRIStAL (UMR 9189) (France)
[□] Universided Carlos III de Madrid (Spain)

***Abstract*—The adaptive multiple importance sampling (AMIS) algorithm is a powerful Monte Carlo tool for Bayesian estimation in intractable models. The uniqueness of this methodology from other adaptive importance sampling (AIS) schemes is in the weighting procedure, where at each iteration of the algorithm, all samples are re-weighted according to the temporal deterministic mixture approach. This re-weighting allows for substantial variance reduction of the AMIS estimator, at the expense of an increased computational cost that grows quadratically with the number of iterations. In this paper, we propose a novel AIS methodology which obtains most of the AMIS variance reduction while improving upon its computational complexity. The proposed method implements an approximate version of the temporal deterministic mixture approach and requires substantially less computation. Advantages are shown empirically through a numerical example, where the novel method is able to attain a desired mean-squared error with much less computation.**

I. Introduction

Importance sampling (IS) methods are widely employed in signal processing to perform approximate Bayesian inference. In particular, they are implemented when the posterior probability density function (pdf) of a random variable of interest, called the target distribution, is complicated (i.e., its moments cannot be analytically computed and generating samples from this pdf is impossible). IS samples from a simpler proposal pdf, assigning weights to them according to the ratio between the target and the proposal. Although the validity of this approach is guaranteed under mild assumptions, the variance of the estimator depends notably on the discrepancy between the shape of the proposal and the target [1], [2].

In order to overcome this problem, substantial effort has been devoted to the design of adaptive IS (AIS) schemes, where the proposal density is updated by learning from all the previously generated samples [3], [4]. Population Monte Carlo (PMC) schemes [5]–[9] and the adaptive population importance samplers (APIS) [10], [11] are two general approaches that combine the proposal adaptation idea with the cooperative use of a cloud of proposal pdfs. In the PMC schemes a population of proposals is updated, e.g., changing their location parameters, by the use of resampling steps [2, Chapter 14], [5], [9]. In other methodologies, such as mixture PMC (M-PMC), all the parameters of a mixture proposal distribution are adapted [6]. Moreover, different types of adaptation have been proposed, for instance, by the use of MCMC outputs [11]–[14]. Some of these techniques consider the application of the so-called deterministic mixture weighting procedure, which provides more efficient IS estimators when several different proposal pdfs are jointly employed [15]–[17].

The adaptive multiple importance sampling (AMIS) algorithm [18] occupies a central place in the literature, due to its simplicity and efficiency. In AMIS, a single proposal is adapted by computing empirical moment estimators, but a *temporal* deterministic mixture approach is employed to build the importance weights [15], [16]. For this reason, the resulting estimators often present smaller variance than the estimators corresponding to other AIS methods [17]. However, this weight construction implies that all the previous proposals must be evaluated at the new samples, and also that the new proposal pdf must be evaluated at all the previous samples, thus yielding an increase in computational cost as the algorithm evolves in time.

In this work, we develop a novel strategy for reducing the computation of the AMIS algorithm. In particular, we formulate an approximation to the temporal deterministic mixture approach by reducing the number of mixture components in the denominator of the importance weights. The approximation is based on the idea that after a certain number of iterations, the discrepancy between subsequently adapted proposal distributions is minimal. We also develop an automated version of the algorithm which detects the number of iterations at which the approximation is appropriate.

II. Problem Formulation

We consider an unknown vector $\mathbf{x} \in \mathbb{R}^{d_x}$ that must be estimated from a set of observed data $\mathbf{y} \in \mathbb{R}^{d_y}$. A statistical model that relates the unknowns and the data is assumed to be known and summarized in the posterior pdf defined as

$$p(\mathbf{x}|\mathbf{y}) = \frac{\ell(\mathbf{y}|\mathbf{x})h(\mathbf{x})}{p(\mathbf{y})} \propto \ell(\mathbf{y}|\mathbf{x})h(\mathbf{x}), \tag{1}$$

where $\ell(\mathbf{y}|\mathbf{x})$ is the likelihood function, $h(\mathbf{x})$ is the prior pdf, and $Z \equiv p(\mathbf{y})$ is the normalization factor. We denote $\pi(\mathbf{x}) \equiv \ell(\mathbf{y}|\mathbf{x})h(\mathbf{x})$. The objective is to approximate the posterior $p(\mathbf{x}|\mathbf{y})$ and moments of the form

$$I = \int g(\mathbf{x})p(\mathbf{x}|\mathbf{y})d\mathbf{x}, \tag{2}$$

V.E. acknowledges support from the *Agence Nationale de la Recherche* of France under PISCES project (ANR-17-CE40-0031-01). M.B thanks the support of the NSF under Award CCF-1617986. The authors thank the Research Computing and Cyberinfrastructure and the Institute for Advanced Computational Science at Stony Brook University for access to the high-performance SeaWulf computing system sponsored by NSF (# 1531492).

where $g(\mathbf{x})$ is an integrable function w.r.t. the target distribution. In this paper, we focus on MC-based sampling [1], [2], and in particular on AIS for the approximation task.

III. Background

A. Importance sampling (IS)

IS is a Monte Carlo methodology for approximating distributions and their moments. In its basic implementation, M samples are simulated from a proposal distribution $q(\mathbf{x})$ as $\mathbf{x}^{(m)} \sim q(\mathbf{x})$ for $m = 1, \dots, M$. An importance weight is assigned to each sample as $w^{(m)} = \frac{\pi(\mathbf{x}^{(m)})}{q(\mathbf{x}^{(m)})}$ for $m = 1, ..., M$. Under mild conditions, IS allows for an approximation of (2) through the the self-normalized IS estimator if Z is unknown,

$$\widehat{I} = \sum_{m=1}^{M} \bar{w}^{(m)} g(\mathbf{x}^{(m)}), \tag{3}$$

where $\bar{w}^{(m)} = \frac{w^{(m)}}{\sum_{j=1}^{M} w^{(j)}}$ are the normalized weights. It is well-known that the variance of (3) augments with the mismatch between the proposal distribution $q(\mathbf{x})$ and $|g(\mathbf{x})|\pi(\mathbf{x})$. For this reason, adaptive mechanisms are often employed.

B. Adaptive importance sampling (AIS)

The standard AIS methodology is an iterative process which provides a gradual evolution of the proposal density, becoming closer and closer to the target pdf. Generally, one considers a parametric proposal density. We denote the vector of the proposal's parameters as $\boldsymbol{\theta}$. The procedure consists of three basic steps: generation of samples from a proposal or set of proposals (sampling), calculation of the importance of each of the samples (weighting), and updating the parameters that define the proposal to obtain the new proposal for the next iteration (adapting). Typically, a standard AIS approach considers a unique proposal with parameters $\boldsymbol{\theta}_t$ (where t represents the t-th iteration of the algorithm) and, with the adaptation, we have $\boldsymbol{\theta}_t \to \boldsymbol{\theta}_\infty$ as $t \to \infty$. Namely, a classical AIS scheme often reaches a stable configuration after a certain number of iterations [3], [4]. The adaptation is often implemented by the use of empirical moment estimators [4], [19].

C. Adaptive Multiple Importance Sampling (AMIS)

In the AMIS algorithm, one proposal is adapted over the iterations. The adaptive procedure consists of estimating the moments of the target with the available set of weighted samples, and fitting the moments of the proposal (e.g., mean and covariance matrix), as in classical AIS. Its key feature is the re-weighting of all the past samples with a *temporal* mixture weight where the whole sequence of proposals is used in the denominator of the weights [18], [20]. AMIS is outlined in Table I. Note that we denote $\boldsymbol{\theta}_j = \{\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\}$.

It is important to notice that: (a) the temporal mixture in the denominator of the weight calculation is incremental (Step 2b., Table I) and at each iteration a new mixand is added with parameters obtained using all samples and weights from previous iterations; and (b) all the previous samples are re-weighted at each iteration, considering the new denominator of the weights (with an additional component).

Algorithm: AMIS

We denote $\boldsymbol{\theta}_j = \{\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\}$.

1. Initialization: Select the initial proposal $q(\mathbf{x}; \boldsymbol{\theta}_1)$.

2. For $t = 1, ..., T$

a. Draw M samples from the current proposal:
$\mathbf{x}_t^{(m)} \sim q(\mathbf{x}; \boldsymbol{\theta}_t), \quad m = 1, \dots, M.$

b. Weighting and re-weighting procedure:
$$w_\tau^{(m)} = \frac{\pi(\mathbf{x}_\tau^{(m)})}{\frac{1}{t}\sum_{j=1}^{t} q(\mathbf{x}_\tau^{(m)}; \boldsymbol{\theta}_j)}, \quad \begin{matrix} m = 1, \dots, M \\ \tau = 1, \dots, t. \end{matrix}$$

c. Normalize the importance weights,
$$\bar{w}_\tau^{(m)} = \frac{w_\tau^{(m)}}{\sum_{j=1}^{t}\sum_{m=1}^{M} w_j^{(m)}}, \quad \begin{matrix} m = 1, \dots, M \\ \tau = 1, \dots, t. \end{matrix}$$

d. Update the proposal parameters,
$$\boldsymbol{\mu}_{t+1} = \sum_{j=1}^{t}\sum_{m=1}^{M} \bar{w}_j^{(m)} \mathbf{x}_j^{(m)},$$
$$\boldsymbol{\Sigma}_{t+1} = \sum_{j=1}^{t}\sum_{m=1}^{M} \bar{w}_j^{(m)} (\mathbf{x}_j^{(m)} - \boldsymbol{\mu}_{t+1})(\mathbf{x}_j^{(m)} - \boldsymbol{\mu}_{t+1})^\intercal$$

3. Return $\{\mathbf{x}_t^{(m)}, w_t^{(m)}\}_{m=1}^{M}$ for $t = 1, ..., T$.

TABLE I: The AMIS algorithm.

AMIS performs very well in different scenarios [4], [18], [20]. However, its computational cost increases with t. Specifically, the number of proposal evaluations is $\mathcal{O}(MT^2)$, which makes AMIS unfeasible after a certain number of iterations.

IV. Proposed Algorithm

We introduce *efficient adaptive multiple importance sampling* (EAMIS), a novel AIS methodology which performs similarly to AMIS, but with a reduced computational cost. The efficiency of the scheme comes at the expense of a bias which vanishes over iterations. The algorithm is shown in Table II.

A. Motivation

An interesting solution to consider for reducing the computational complexity of the re-weighting procedure is to approximate the full temporal mixture using a mixture distribution with a small number of components (reduced mixture). To avoid additional computation, it is also desirable that we make this approximation using only mixture components that we have already evaluated. Consider that at iteration $t > K$ we would like to update the weights of samples from the previous iteration. Instead of evaluating the new proposal $q(\mathbf{x}; \boldsymbol{\theta}_t)$ for all previous samples (as in Step 2b., Table I), we would like to make the following approximation of the temporal mixture,

$$\frac{1}{t}\sum_{j=1}^{t} q(\mathbf{x}; \boldsymbol{\theta}_j) \approx \sum_{j=1}^{K} \alpha_j q(\mathbf{x}; \boldsymbol{\theta}_j). \tag{4}$$

The goal is to choose the weights α_j (for $j = 1, ..., K$) to justify the above approximation for all $t > K$.

Algorithm: EAMIS

We denote $\boldsymbol{\theta}_j = \{\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\}$.

1. Initialization: Select the initial proposal $q(\mathbf{x}; \boldsymbol{\theta}_1)$.

2. For $t = 1, ..., T$

a. Draw M samples from the current proposal: $\mathbf{x}_t^{(m)} \sim q(\mathbf{x}; \boldsymbol{\theta}_t), \quad m = 1, \ldots, M.$

b. Weighting and re-weighting procedure:

if $(t < K)$ **then**:

$$w_\tau^{(m)} = \frac{\pi(\mathbf{x}_\tau^{(m)})}{\frac{1}{t}\sum_{j=1}^{t} q(\mathbf{x}_\tau^{(m)}; \boldsymbol{\theta}_j)}, \qquad \begin{matrix} m = 1, \ldots, M \\ \tau = 1, \ldots, t. \end{matrix}$$

else:

$$w_\tau^{(m)} = \frac{\pi(\mathbf{x}_\tau^{(m)})}{\frac{1}{t}\sum_{j=1}^{K-1} q(\mathbf{x}_\tau^{(m)}; \boldsymbol{\theta}_j) + \frac{t-K+1}{t} q(\mathbf{x}_\tau^{(m)}; \boldsymbol{\theta}_{\ell^*})},$$

$$m = 1, \ldots, M, \;\; \tau = 1, \ldots, t, \;\; \ell^* = \max(\tau, K).$$

c. Normalize the importance weights (see Table I).

d. Update proposal parameters (see Table I).

3. Return $\{\mathbf{x}_t^{(m)}, w_t^{(m)}\}_{m=1}^{M}$ for $t = 1, ..., T$.

TABLE II: The EAMIS algorithm.

B. Algorithm Summary

Suppose that we run the AMIS algorithm for K iterations. In this scenario, we have generated samples $\mathbf{x}_t^{(m)} \sim q(\mathbf{x}; \boldsymbol{\theta}_t)$ for $m = 1, \ldots, M$ and $t = 1, \ldots, K$. Following the standard procedure of AMIS, we have a set of weights given by,

$$w_\tau^{(m)} = \frac{\pi(\mathbf{x}_\tau^{(m)})}{\frac{1}{K}\sum_{j=1}^{K} q(\mathbf{x}_\tau^{(m)}; \boldsymbol{\theta}_j)}, \tag{5}$$

for $m = 1, \ldots, M$ and $\tau = 1, \ldots, K$. Now suppose that we generate $\mathbf{x}_{K+1}^{(m)} \sim q(\mathbf{x}; \boldsymbol{\theta}_{K+1})$ for $m = 1, \ldots, M$. According to the AMIS procedure, we need to evaluate $q(\mathbf{x}; \boldsymbol{\theta}_{K+1})$ for all previous samples in order to reweigh the previous samples. Assuming that $\|\boldsymbol{\mu}_{K+1} - \boldsymbol{\mu}_K\| < \gamma_1$ and $\|\boldsymbol{\Sigma}_{K+1} - \boldsymbol{\Sigma}_K\| < \gamma_2$ for $0 < \gamma_1, \gamma_2 << 1$, where $\|\cdot\|$ denotes a norm (e.g., L_2 or Frobenius), we can justify the following approximation: $q(\mathbf{x}; \boldsymbol{\theta}_{K+1}) \approx q(\mathbf{x}; \boldsymbol{\theta}_K)$. Intuitively, this means that evaluating $q(\mathbf{x}; \boldsymbol{\theta}_{K+1})$ is approximately the same as evaluating $q(\mathbf{x}; \boldsymbol{\theta}_K)$ for all previous samples and weights. However, we have already evaluated $q(\mathbf{x}; \boldsymbol{\theta}_K)$ to apply the temporal deterministic mixture approach at iteration K in (5). Then, re-weighing the previous samples is equivalent to updating the mixture weight for $q(\mathbf{x}; \boldsymbol{\theta}_K)$ as follows,

$$w_\tau^{(m)} = \frac{\pi(\mathbf{x}_\tau^{(m)})}{\frac{1}{K+1}\sum_{j=1}^{K-1} q(\mathbf{x}_\tau^{(m)}; \boldsymbol{\theta}_j) + \frac{2}{K+1} q(\mathbf{x}_\tau^{(m)}; \boldsymbol{\theta}_K)}, \tag{6}$$

for $m = 1, \ldots, M$ and $\tau = 1, \ldots, K$. For the weights $\{w_{K+1}^{(m)}\}_{m=1}^{M}$, we use $q(\mathbf{x}; \boldsymbol{\theta}_1), \ldots, q(\mathbf{x}; \boldsymbol{\theta}_{K-1})$ and the newly adapted $q(\mathbf{x}; \boldsymbol{\theta}_{K+1})$ for the weight computation. In essence, we can summarize the re-weighting procedure of the EAMIS algorithm for iterations $t \geq K$ with the following:

$$w_\tau^{(m)} = \frac{\pi(\mathbf{x}_\tau^{(m)})}{\frac{1}{t}\sum_{j=1}^{K-1} q(\mathbf{x}_\tau^{(m)}; \boldsymbol{\theta}_j) + \frac{t-K+1}{t} q(\mathbf{x}_\tau^{(m)}; \boldsymbol{\theta}_{\ell^*})}, \tag{7}$$

for $m = 1, \ldots, M$ and $\tau = 1, \ldots, t$. Here ℓ^* depends on τ and is defined as $\ell^* = \max(\tau, K)$. Therefore, after iteration K we reweigh samples generated in iterations $\tau = 1, \ldots, K$ by updating the contribution of $q(\mathbf{x}; \boldsymbol{\theta}_K)$, while for $\tau > K$ we update the contribution of the proposals used to generate samples in that iteration. This has interesting theoretical guarantees. Mainly, for the samples generated at iteration $t \geq K$,

$$w_t^{(m)} \to \frac{\pi\big(\mathbf{x}_t^{(m)}\big)}{q\big(\mathbf{x}_t^{(m)}; \boldsymbol{\theta}_t\big)} \quad \text{as} \quad t \to \infty, \tag{8}$$

which are just the standard IS weights. This result will provide an important insight related to the convergence of the methodology that will be studied in the extension of this work.

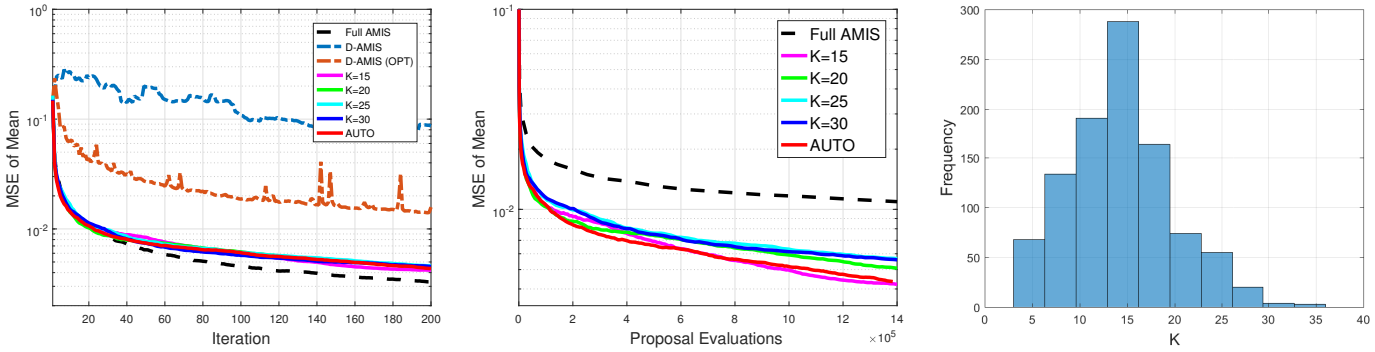
Other variants of this approximation could be designed, however, the assumption must hold that for all $t > K$ $q(\mathbf{x}; \boldsymbol{\theta}_t) \approx q(\mathbf{x}; \boldsymbol{\theta}_K)$. Note that, for a great enough K, this assumption is fulfilled since the adaptation is performed as in a standard AIS scheme where $\boldsymbol{\theta}_t \to \boldsymbol{\theta}_\infty$ for $t \to \infty$. We emphasize that the key contribution is that by leveraging this approximation, we can approximate the weights from the previous iterations of the algorithm with a substantial decrease in the computational cost of evaluating proposals.

C. Discussion on Computational Complexity

The number of proposal evaluations required at each iteration of the AMIS algorithm is $\mathcal{O}(M(2t-1))$ for $t = 1, \ldots, T$. The total complexity is then $\mathcal{O}(MT^2)$. For the proposed method, the per iteration complexity is the same as AMIS for the first K iterations. For $t > K$, we exploit the fact that besides re-weighting the temporal mixtures for the previous iterations, we only need to evaluate K new proposals at each iteration. Then the total complexity of EAMIS is $\mathcal{O}(MK(T-K) + \sum_{t=1}^{K} M(2t-1))$. We can summarize this complexity as $\mathcal{O}(MKT)$. This gives the ability to choose K to achieve a desired algorithm complexity. We note that if K is chosen too small, then there is risk that the approximation $q(\mathbf{x}; \boldsymbol{\theta}_t) \approx q(\mathbf{x}; \boldsymbol{\theta}_K)$ may not hold. In the following section, we discuss a strategy to automatically choose K such that the approximation is satisfied with limited computation.

D. Automatically Choosing K

Here we consider a mechanism automatically selecting K. Our goal is to detect the iteration number in which the proposal adaptations become minuscule, i.e., the value of K for which $q(\mathbf{x}; \boldsymbol{\theta}_{K+1}) \approx q(\mathbf{x}; \boldsymbol{\theta}_K)$. To do this, at each iteration we can consider a discrepancy measure $\mathcal{D}(\cdot)$ between the current proposal and the newly adapted proposal. If $\mathcal{D}(q(\mathbf{x}; \boldsymbol{\theta}_{t+1}), q(\mathbf{x}; \boldsymbol{\theta}_t)) < \epsilon$, then we set $K = t$. There are many discrepancy measures that can be chosen. For example, we can simply use the L_2 norm of the difference between the means $\|\boldsymbol{\mu}_{t+1} - \boldsymbol{\mu}_t\|_2$. When the proposal distributions are Gaussian, we can directly evaluate the Kullback-Leibler divergence (KLD) using a closed-form expression. We note that the KLD is not a true distance measure since it is asymmetrical, however, it does take into account the covariance structure of each distribution. We also point out that evaluation of the KLD requires both the determinant of the covariance matrix and its inverse (precision matrix), which will include an additional computational cost that will depend on the dimension d_x.



(a) MSE versus iteration number. (b) MSE versus proposal evaluations. (c) Histogram of K over 1000 realizations.

Fig. 1: Summary of results from the numerical example.

V. Numerical Examples

A. 2-Dimensional Banana Target

Suppose that $\mathbf{x} = [x_1, x_2]^\intercal \in \mathbb{R}^2$. For this example, the target distribution of interest is a two-dimensional banana-shaped distribution,

$$\psi(\mathbf{x}) = \exp\left(-\frac{1}{2\eta_1^2}\left(4 - Bx_1 - x_2^2\right)^2 - \frac{x_1^2}{2\eta_2^2} - \frac{x_2^2}{2\eta_3^2}\right). \quad (9)$$

We note that the target $\pi(\mathbf{x}) \propto \psi(\mathbf{x})$. For our simulations, we set the parameters as follows: $\eta_1 = 4$, $\eta_2 = 3.5$, $\eta_3 = 3.5$, and $B = 10$. Through numerical integration over a grid, we determined the normalizing constant and mean of this target as $Z \approx 7.9978$ and $\mathbb{E}_\pi[\mathbf{X}] \approx [-0.4843, 0]^\intercal$ respectively. We simulated the original AMIS algorithm [18], two variants of the discarding AMIS (D-AMIS) approach in [21] (one which chooses a heuristic discarding rate and one which chooses a discarding rate to optimize effective sample size), and our novel EAMIS approach. For each methodology, we initialized the mean as $\boldsymbol{\mu}_1 \sim \mathcal{U}([-5, -2]^2)$ and set the initial covariance matrix as $\boldsymbol{\Sigma}_1 = 5\mathbb{I}_2$. For the EAMIS methodology, we tested $K \in \{15, 20, 25, 30\}$ and also a variant with automatic selection of K. For the automated EAMIS, our discrepancy measure was the L_2 norm of the difference of the means, i.e., $\|\boldsymbol{\mu}_t - \boldsymbol{\mu}_{t-1}\|_2$ with threshold $\epsilon = 0.005$. The goal was to estimate the mean of the target distribution, $\mathbb{E}_\pi[\mathbf{X}]$. We averaged our results over 1000 MC simulations.

Figure 1a plots the evolution of the average mean-squared error (MSE) over iterations. The performance of our algorithm is similar to that of AMIS. The methods proposed in [21] perform poorly in this example, even when the optimized variant is used. It is interesting to point out that the discarding step in D-AMIS results in 'noisy' performance, so despite the D-AMIS methods have smallest computational complexity, the performance does not compare to AMIS or to the proposed methodology. Figure 1b highlights the advantage of the proposed method over the AMIS algorithm. We can see that EAMIS is able to achieve lower MSE at a faster computational rate. In other words, AMIS requires many more evaluations of the proposal distribution in order to achieve the same or better performance as EAMIS. We also plot a histogram of the chosen K in the automated version of the proposed algorithm in Figure 1c. We can see that, depending on the specific initialization of the proposal distribution, the algorithm results in a different value of K on each run. This helps to eliminate the concern of choosing an improper value of K, which would cause a negative effect in the performance of EAMIS.

B. 10-Dimensional Banana Target

We extended the previous example to one of higher dimension, such that $\mathbf{x} \in \mathbb{R}^{10}$. The target of interest becomes

$$\pi(\mathbf{x}) \propto \psi([x_1, x_2]^\intercal) \prod_{j=3}^{10} \mathcal{N}(x_j; 0, 1), \quad (10)$$

where $\mathcal{N}(x_j; 0, 1)$ denotes the pdf of a standard normal distribution. For this example, $Z \approx 7.9978$ and target mean $\mathbb{E}_\pi[\mathbf{X}] \approx [-0.4843, 0, \ldots, 0]^\intercal$. We simulated AMIS and the automated EAMIS using $M = 2000$. Each method ran until 10^7 proposal evaluations were reached. The error metric utilized was average MSE for the estimation of $\mathbb{E}_\pi[\mathbf{X}]$ and the mean-absolute error (MAE) for the estimation of Z. Results were averaged over 1000 MC simulations and are shown in Table III. It is clear from the results that the EAMIS technique outperforms AMIS. For AMIS to achieve comparable performance, many more proposal evaluations would be needed, emphasizing the key advantage of EAMIS.

	MSE($\mathbb{E}_\pi[\mathbf{X}]$)	MAE (Z)
AMIS	0.0174	0.7853
EAMIS	**0.0061**	**0.2538**

TABLE III: Results for 10-dimensional banana target.

VI. Conclusions

In this work, we proposed a novel adaptive importance sampling (AIS) scheme which improves upon the computational disadvantages of the adaptive multiple importance sampling (AMIS) algorithm. The method employs the re-weighting procedure of AMIS by approximating the temporal deterministic mixture without the need of evaluating new proposals for samples drawn in past iterations. Furthermore, we developed an automated version of the scheme, which robustly detects when the approximation of the temporal mixture is valid. Simulation results indicate that the novel method outperforms AMIS as the number of proposal evaluations to achieve a desired mean-squared error (MSE) is much less for the proposed method than for AMIS.

REFERENCES

[1] J. S. Liu, *Monte Carlo Strategies in Scientific Computing*, Springer, 2004.

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

[3] L. Martino, V. Elvira, D. Luengo, and F. Louzada, "Adaptive population importance samplers: A general perspective," *IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM)*, pp. 1–5, 2016.

[4] 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, July 2017.

[5] 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.

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

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

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

[9] V. Elvira, L. Martino, D. Luengo, and M. F. Bugallo, "Improving population monte carlo: Alternative weighting and resampling schemes," *Signal Processing*, vol. 131, no. Supplement C, pp. 77 – 91, 2017.

[10] 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, Aug 2015.

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

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

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

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

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

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

[17] V. Elvira, L. Martino, D. Luengo, and M. F. Bugallo, "Generalized multiple importance sampling," *(to appear) Statistical Science*, pp. 1–42, 2018.

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

[19] G. H. Givens and A. E. Raftery, "Local adaptive importance sampling for multivariate densities with strong nonlinear relationships," *Journal of the American Statistical Association*, vol. 91, no. 443, pp. 132–141, 1996.

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

[21] S. Thijssen and H. Kappen, "Consistent adaptive multiple importance sampling and controlled diffusions," *arXiv preprint arXiv:1803.07966*, 2018.