AN ADAPTIVE POPULATION IMPORTANCE SAMPLER

Luca Martino**, Víctor Elvira*†*, David Luengo*‡*, Jukka Corander**

* Dep. of Mathematics and Statistics, University of Helsinki, 00014 Helsinki (Finland).
† Dep. of Signal Theory and Communic., Universidad Carlos III de Madrid, 28911 Leganés (Spain).
‡ Dep. of Circuits and Systems Engineering, Universidad Politécnica de Madrid, 28031 Madrid (Spain).

ABSTRACT

Monte Carlo (MC) methods are widely used in signal processing, machine learning and communications for statistical inference and stochastic optimization. A well-known class of MC methods is composed of importance sampling and its adaptive extensions (e.g., population Monte Carlo). In this work, we introduce an adaptive importance sampler using a population of proposal densities. The novel algorithm provides a global estimation of the variables of interest iteratively, using all the samples generated. The cloud of proposals is adapted by learning from a subset of previously generated samples, in such a way that local features of the target density can be better taken into account compared to single global adaptation procedures. Numerical results show the advantages of the proposed sampling scheme in terms of mean absolute error and robustness to initialization.

Index Terms— Monte Carlo methods, adaptive importance sampling, population Monte Carlo, iterative estimation.

1. INTRODUCTION

Monte Carlo methods are widely used in signal processing and communications [1, 2]. Importance sampling (IS) [3, 4] is a well-known Monte Carlo (MC) methodology to compute integrals involving a complicated multidimensional target probability density function (pdf), $\pi(\mathbf{x})$ with $\mathbf{x} \in \mathbb{R}^n$, efficiently. The IS technique draws samples from a simple proposal pdf, $q(\mathbf{x})$, assigning weights to them according to the ratio between the target and the proposal, i.e., $w(\mathbf{x}) = \frac{\pi(\mathbf{x})}{q(\mathbf{x})}$. However, although the validity of this approach is guaranteed under mild assumptions, the variance of the estimator depends critically on the discrepancy between the shape of the proposal and the target. For this reason, Markov Chain Monte Carlo (MCMC) methods are usually preferred for large dimensional applications [5, 6, 7, 8].

In order to solve this issue, several works are devoted to the design of adaptive IS (AIS) schemes [4], where the proposal density is updated by learning from all the previously generated samples. The Population Monte Carlo (PMC) [9] and the Adaptive Multiple Importance Sampling (AMIS) [10] methods are two general schemes that combine the proposal adaptation idea with the cooperative use of a population of proposal pdfs. In PMC, a cloud of proposals is updated using propagations and resampling steps [4, Chapter 14]. In AMIS, a single proposal is adapted following a standard adaptive IS scheme, but the sequence of all the previous proposals is used to build the global estimator (implying that all the previous proposals must be evaluated at the new samples, thus leading to an increase in computational cost as the algorithm evolves).

In this work, we introduce a novel population scheme, *adaptive population importance sampling* (APIS). APIS draws samples from different proposal densities at each iteration, weighting them according to the so-called *deterministic mixture* approach, proposed in [11, 12] for a fixed (i.e., non-adaptive) setting. At each iteration, APIS computes iteratively a *global* IS estimate, taking into account all the generated samples up to that point. The main difference w.r.t. AMIS and PMC lies in its more streamlined adaptation procedure. APIS starts with a cloud of N proposals initialized randomly or according to the prior information available. The algorithm is then divided into groups of T_a iterations (so called *epochs*), where the proposals are kept fixed and T_a samples are drawn from each one. At the end of every epoch, the T_a samples drawn from each proposal are used to update its parameters (using *partial IS estimators*). APIS does not require resampling steps to prevent the degeneracy of the mixture (as in PMC) and its computational cost does not increase with the iteration number (as in AMIS).

For the sake of simplicity, in this work we focus on a specific implementation with Gaussian proposal pdfs, whose means are updated according to the partial IS estimators of the expected value of the target, given T_a samples from each Gaussian. In this way, APIS takes advantage of one of the drawbacks of a standard IS method, since each proposal is able to extract specific and localized features of the target efficiently. Thus, one proposal can describe a specific region, while the remaining proposals explore other parts of the state space. Numerical results show that APIS improves the performance of a standard non-adaptive multiple importance sampler regardless of the initial conditions and parameters.

This work has been supported by the Spanish government's projects COMONSENS (CSD2008-00010), ALCIT (TEC2012-38800-C03-01), DISSECT (TEC2012-38058-C03-01) and COMPREHENSION (TEC2012-38883-C02-01), and by the ERC grant 239784 and AoF grant 251170.

978-1-4799-2893-4/14/$31.00 ©2014 IEEE

2. PROBLEM STATEMENT

In many applications, we are interested in inferring a variable of interest given a set of observations or measurements. Let us consider the variable of interest, $\mathbf{x} \in \mathbb{R}^n$, and let $\mathbf{y} \in \mathbb{R}^d$ be the observed data. The posterior pdf is then

$$p(\mathbf{x}|\mathbf{y}) = \frac{\ell(\mathbf{y}|\mathbf{x})g(\mathbf{x})}{Z(\mathbf{y})} \propto \ell(\mathbf{y}|\mathbf{x})g(\mathbf{x}), \qquad (1)$$

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

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

Our goal is computing efficiently some moment of $\mathbf{x}$, i.e., an integral measure w.r.t. the target pdf,

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

where $Z = \int_{\mathcal{X}} \pi(\mathbf{x})d\mathbf{x}$.

3. THE APIS ALGORITHM

The adaptive population importance sampling (APIS) algorithm tries to estimate Z and I by drawing samples from a population of adaptive proposals. For the sake of simplicity, here we only consider a population of Gaussian proposal pdfs with fixed covariance matrices and we adapt only the means. However, the underlying idea is more general: many kinds of proposals could be used, including mixtures of different types of proposals. Furthermore, other parameters (e.g., the covariance matrices or any other shape/scale parameters) could also be updated.[1]

3.1. Algorithm

The APIS algorithm is summarized below.

1. **Initialization:** Set $t = 1$, $m = 0$, $\hat{I}_0 = 0$ and $L_0 = 0$. Choose N *normalized* Gaussian proposal pdfs,

$$q_i^{(0)}(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_i^{(0)}, \mathbf{C}_i), \qquad i = 1, \ldots, N,$$

   with mean vectors $\boldsymbol{\mu}_i^{(0)}$ and covariance matrices $\mathbf{C}_i$ ($i = 1, \ldots, N$). Select the number of iterations per epoch, $T_a \geq 2$, and the total number of iterations, $T = MT_a$, with $M \leq \frac{T}{2} \in \mathbb{Z}^+$ denoting the number of adaptation epochs. Set also $\boldsymbol{\eta}_i = \mathbf{0}$ and $W_i = 0$ for $i = 1, \ldots, N$.

2. **IS steps:**

   (a) Draw $\mathbf{z}_i \sim q_i^{(m)}(\mathbf{x})$ for $i = 1, \ldots, N$.

   (b) Compute the importance weights,

$$w_i = \frac{\pi(\mathbf{z}_i)}{\frac{1}{N}\sum_{j=1}^{N} q_j^{(m)}(\mathbf{z}_i)}, \qquad i = 1, \ldots, N, \qquad (4)$$

   and normalize them,

$$\bar{w}_i = \frac{w_i}{S}, \qquad (5)$$

   with $S = \sum_{j=1}^{N} w_j$.

3. **Iterative IS estimation:** Calculate the "current" estimate of I,

$$\hat{J}_t = \sum_{i=1}^{N} \bar{w}_i f(\mathbf{z}_i), \qquad (6)$$

   and the *global estimate*, using the recursive formula

$$\hat{I}_t = \frac{1}{L_{t-1} + S}\left(L_{t-1}\hat{I}_{t-1} + S\hat{J}_t\right), \qquad (7)$$

   where $L_t = L_{t-1} + S$. Note that $\hat{Z}_t = \frac{1}{Nt}L_t$.

4. **Learning:**

   (a) Compute

$$\rho_i = \frac{\pi(\mathbf{z}_i)}{q_i^{(m)}(\mathbf{z}_i)}, \qquad i = 1, \ldots, N. \qquad (8)$$

   (b) Calculate the empirical means,

$$\boldsymbol{\eta}_i = \frac{1}{W_i + \rho_i}\left(W_i\boldsymbol{\eta}_i + \rho_i\mathbf{z}_i\right), \qquad (9)$$

   and set $W_i = W_i + \rho_i$ for $i = 1, \ldots, N$.

5. **Proposal adaptation:** If $t = kT_a$ ($k = 1, 2, \ldots, M$):

   (a) Adapt the proposals, moving them to the locations corresponding to their empirical means, i.e., set

$$\boldsymbol{\mu}_i^{(m+1)} = \boldsymbol{\eta}_i, \qquad i = 1, \ldots, N, \qquad (10)$$

   and $q_i^{(m+1)} = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_i^{(m+1)}, \mathbf{C}_i)$.

   (b) "Refresh memory" by setting $\boldsymbol{\eta}_i = \mathbf{0}$ and $W_i = 0$ for $i = 1, \ldots, N$. Set also $m = m + 1$.

6. **Stopping rule:** The simplest possibility is: If $t < T$, set $t = t + 1$ and repeat from step 2. Otherwise, end.

7. **Outputs:** Return the estimate of the desired integral,

$$\hat{I}_T \approx I = \frac{1}{Z}\int_{\mathcal{X}} f(\mathbf{x})\pi(\mathbf{x})d\mathbf{x}, \qquad (11)$$

   as well as the normalizing constant of the target pdf,

$$\hat{Z}_T \approx Z = \int_{\mathcal{X}} \pi(\mathbf{x})d\mathbf{x}. \qquad (12)$$

The final locations of the Gaussians (i.e., their means, $\boldsymbol{\mu}_i^{(M)}$ for $i = 1, \ldots, N$) could also be used to estimate the locations of the modes of $\pi(\mathbf{x})$.

[1] The joint adaptation of different types of parameters is more delicate, so we leave it for a future work.

3.2. Remarks and observations

In this section, we provide some important remarks on several aspects of the APIS algorithm:

1. All the different proposal pdfs should be normalized to provide a correct IS estimation.

2. The global estimators, $\hat{I}_T$ and $\hat{Z}_T$, are iteratively obtained by an importance sampling approach using NT total samples drawn (in general) from NT different proposals: N initial proposals chosen by the user, and $N(T-1)$ proposals adapted by the algorithm.

3. Different stopping rules can be applied to ensure that the global estimators produce the desired degree of accuracy, in terms of Monte Carlo variability. For instance, one possibility is taking into account the variation of the estimate over time. In this case, the algorithm could be stopped at any iteration $t^* < T$, since an IS approach does not have the convergence issues ("burn-in" period) appearing in MCMC methods.

Moreover, let us observe that the algorithm works on two different time scales:

1. At each iteration ($t = 1, \ldots, T = MT_a$), APIS computes the "current" estimate of the desired integral, $\hat{J}_t$, and updates recursively the global estimates of the desired integral and the normalizing constant, $\hat{I}_t$ and $\hat{Z}_t$ respectively.

2. At the transition iterations between two epochs ($t = mT_a$ with $m = 1, \ldots, M$), the parameters of the proposals, $\boldsymbol{\mu}_i^{(m)}$ for $1 \le i \le N$, are updated.

Considering only the transitions (i.e., $t = mT_a$), APIS can be seen as a parallel implementation of N different adaptive IS methods using $T_a = \frac{T}{M} \ge 2$ samples to adapt the proposal pdfs and providing a single global estimation. Thus, in the previous description the index t could be removed. Indeed, *within an epoch* the proposals do not change, so we could draw T_a i.i.d. samples directly from each proposal and then adapt the proposals using these samples. However, we prefer to maintain the previous description to emphasize the fact that the accuracy of the estimator can be tested at each iteration t, and that the algorithm could be stopped at any time.

3.3. Black-box implementation

As in any other Monte Carlo technique, the performance of APIS depends on the initialization, although this sensitivity is reduced w.r.t. a standard IS approach, as illustrated in the simulations. Hence, if some prior information about the target is available, it should be used to choose the initial parameters. However, if no prior information is available, a possible *black-box* implementation of APIS is the following. **(a)** Select randomly N_μ different means in order to cover as much as possible of the target's domain, $\mathcal{X} \subseteq \mathbb{R}^n$. **(b)** For each mean, choose N_σ different covariance matrices, implying that the total number of different proposals is $N = N_\mu N_\sigma$.

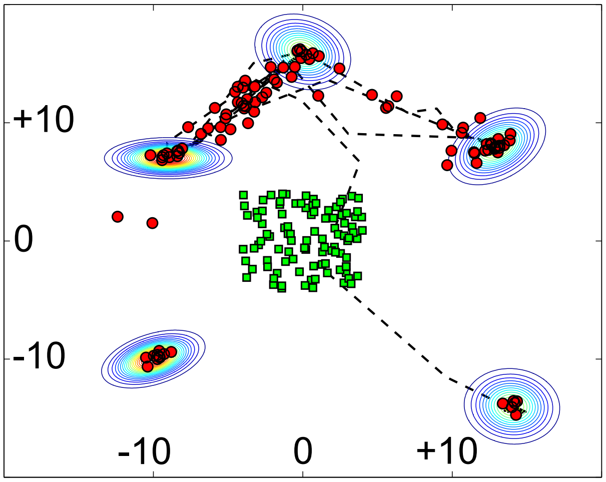


Fig. 1. Contour plot of the target $\pi(\mathbf{x})$, the initial $\boldsymbol{\mu}_i^{(0)}$ (squares) and the final $\boldsymbol{\mu}_i^{(T)}$ (circles) locations of the means of the proposals for a single run of APIS ($\sigma_i = 5$, $N = 100$, $M = 40$, $T = 2000$). The trajectories of two means in the sample population are depicted in dashed line.

4. NUMERICAL RESULTS

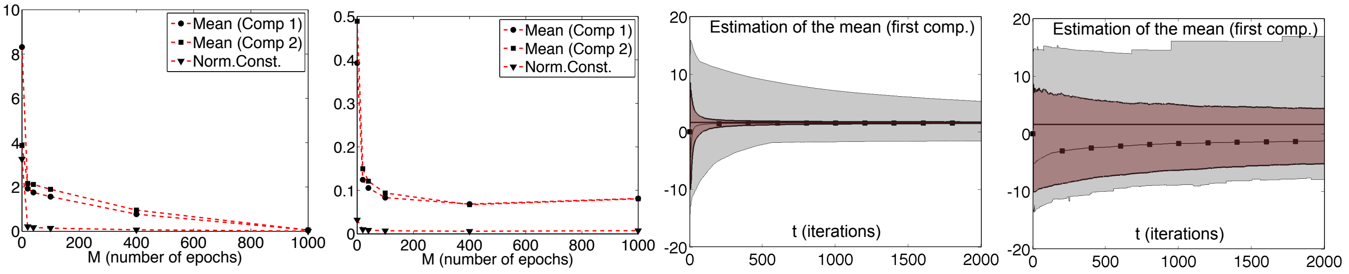
For the simulations, we consider a bivariate multimodal target pdf, which is itself a mixture of 5 Gaussians, i.e.,

$$\pi(\mathbf{x}) = \frac{1}{5}\sum_{i=1}^{5} \mathcal{N}(\mathbf{x}; \boldsymbol{\nu}_i, \boldsymbol{\Sigma}_i), \quad \mathbf{x} \in \mathbb{R}^2, \qquad (13)$$

with means $\boldsymbol{\nu}_1 = [-10, -10]^\top$, $\boldsymbol{\nu}_2 = [0, 16]^\top$, $\boldsymbol{\nu}_3 = [13, 8]^\top$, $\boldsymbol{\nu}_4 = [-9, 7]^\top$, $\boldsymbol{\nu}_5 = [14, -14]^\top$, and covariance matrices $\boldsymbol{\Sigma}_1 = [2,\ 0.6; 0.6,\ 1]$, $\boldsymbol{\Sigma}_2 = [2,\ -0.4; -0.4,\ 2]$, $\boldsymbol{\Sigma}_3 = [2,\ 0.8; 0.8,\ 2]$, $\boldsymbol{\Sigma}_4 = [3,\ 0; 0,\ 0.5]$ and $\boldsymbol{\Sigma}_5 = [2,\ -0.1; -0.1,\ 2]$. Fig. 1 shows a contour plot of $\pi(\mathbf{x})$.

We apply APIS with $N = 100$ Gaussian proposals to estimate the mean (true value $[1.6, 1.4]^\top$) and normalizing constant (true value 1) of the target. We choose deliberately a "bad" initialization of the initial means, to test the robustness of the algorithm and its ability to improve the corresponding *static* (i.e., non-adaptive) IS approach. Specifically, the initial means are selected uniformly within a rectangle, $\boldsymbol{\mu}_i^{(0)} \sim \mathcal{U}([-4, 4] \times [-4, 4])$ for $i = 1, \ldots, N$. A single realization of $\boldsymbol{\mu}_i^{(0)}$ is depicted by the squares in Fig. 1.

Initially we use the same isotropic covariance matrix, $\mathbf{C}_i^{(0)} = \sigma^2 \mathbf{I}_2$, for every proposal. We test different values of $\sigma \in \{0.5, 1, 2, 3, 5, 7, 10, 70\}$, to gauge the performance of APIS. Then we also try different non-isotropic diagonal covariance matrices, $\mathbf{C}_i^{(0)} = \text{diag}(\sigma_{i,1}^2, \sigma_{i,2}^2)$, where



(a) (b) (c) (d)

Fig. 2. **(a)-(b)**: Mean absolute error in the estimation of the mean and normalizing constant of $\pi(\mathbf{x})$, averaged over 2000 runs as function of M (number of epochs) for **(a)** $\sigma = 2$ and **(b)** $\sigma = 5$. **(c)-(d)**: Estimate of the first component of the mean as a function of the iterations t for $\sigma = 3$, **(c)** $M = 400$ and **(d)** without adaptation ($M = 1$). The solid lines depict the true mean value (1.6), and the darker and lighter areas show the range of 90% and 100% of the empirical probability mass, respectively.

Epochs \ Scale par.	$\sigma = 0.5$	$\sigma = 1$	$\sigma = 2$	$\sigma = 3$	$\sigma = 5$	$\sigma = 7$	$\sigma = 10$	$\sigma = 70$	$\sigma_{i,j} \sim \mathcal{U}([1, 10])$
$M = 1$ ($T_a = T$)	5.3566	6.8373	8.3148	3.6428	0.3926	0.1326	**0.0886**	0.3376	0.2048
$M = 20$ ($T_a = 100$)	4.6089	3.5248	1.9265	0.9083	0.1244	0.0910	0.0908	0.3397	0.0837
$M = 40$ ($T_a = 50$)	4.0862	3.3079	1.7518	0.7125	0.1056	0.0863	0.0940	**0.3318**	0.0689
$M = 100$ ($T_a = 20$)	3.7727	3.2009	1.5619	0.5776	0.0832	**0.0822**	0.0961	0.3441	0.0593
$M = 400$ ($T_a = 5$)	3.5577	2.6161	0.7708	0.1464	**0.0685**	0.0846	0.0972	0.3539	**0.0535**
$M = 1000$ ($T_a = 2$)	**2.9543**	**0.9967**	**0.0550**	**0.0636**	0.0814	0.0945	0.1102	0.3594	0.0700

Table 1. Mean absolute error in the estimation of the mean of the target (first component), averaged over 2000 runs, for different values of σ and number of epochs, M; $M = 1$ corresponds to a non-adaptive IS method, whereas $M = \frac{T}{2} = 1000$ is the maximum number of epochs possible for $T = 2000$. The best results for each value of σ are highlighted in bold-face.

$\sigma_{i,j} \sim \mathcal{U}([1, 10])$ for $j \in \{1, 2\}$ and $i = 1, \ldots N$. We set $T = 2000$ and $T_a \in \{2, 5, 20, 50, 100\}$, i.e., $M = \frac{T}{T_a} \in \{20, 40, 100, 400, 1000\}$. We also consider $M = 1$, which corresponds to a standard IS technique with multiple proposals and no adaptation. All the results are averaged over 2000 independent experiments.

Fig. 1 shows also the final locations of the means, $\boldsymbol{\mu}_i^{(T)}$, in one run with $\sigma = 5$ using circles. Furthermore, the trajectories of two means in the sample population are depicted in dashed line. Note that a random walk among three modes of the target is induced in one of them, whereas the other converges to the mode that is further away from the origin. Table 1 shows the mean absolute error (MAE) in the estimation of the first component of the mean: APIS always outperforms the non-adaptive standard IS procedure, with the only exception of $\sigma = 10$, where APIS has a negligibly larger error. Figs. 2(a)–(b) illustrate the evolution of the MAE w.r.t. M for $\sigma = 2$ and $\sigma = 5$ respectively, whereas Figs. 2 (c)-(d) show the estimate of the first component of the mean vs. the iteration step t for a case with adaptation ($\sigma = 3$ and $M = 400$) and a case with no adaptation ($\sigma = 3$ and $M = 1$).

5. CONCLUSIONS AND FUTURE LINES

We have introduced a novel adaptive population importance sampling (APIS) algorithm, which is based on applying importance sampling (IS) principles to a population of adaptive proposal pdfs. Compared to other techniques, APIS has a simpler adaptation procedure (based only on partial IS estimations) and could be easily implemented in a parallel and/or a distributed fashion.

Although the APIS scheme is quite general, here we have focused on a specific implementation with Gaussian proposal pdfs, adapting their means. Our experiments have shown that APIS reduces the dependence on the choice of the parameters of the proposal. Indeed, the proposed adaptation procedure almost always improves the results w.r.t. the corresponding standard non-adaptive IS method, regardless of the variances chosen initially. The results suggest that smaller scaling parameters benefit more from a more frequent adaptation. Such an inverse relationship between the variance of the proposals and frequency of adaptation is expected to hold also more generally in a family of adaptive sampling schemes similar to APIS (e.g., if the proposals were mixture densities themselves).

An interesting open issue is whether optimal adaptation schemes could be identified under particular conditions. Also, it would be interesting to explore in detail how the geometry of the target density does in general influence the rate and trajectories of proposal movements. The joint update of scale and shape parameters and interacting adaptation schemes will also be considered in future work.

6. REFERENCES

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

[2] A. Doucet and X. Wang, "Monte Carlo methods for signal processing," *IEEE Signal Processing Magazine*, vol. 22, no. 6, pp. 152–170, Nov. 2005.

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

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

[5] J. Kotecha and Petar M. Djurić, "Gibbs sampling approach for generation of truncated multivariate gaussian random variables," *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 3, pp. 1757–1760, 15–19 Mar. 1999.

[6] W. J. Fitzgerald, "Markov chain Monte Carlo methods with applications to signal processing," *Signal Processing*, vol. 81, no. 1, pp. 3–18, January 2001.

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

[8] D. Luengo and L. Martino, "Fully adaptive Gaussian mixture Metropolis-Hastings algorithm," *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 6148–6152, 26–31 May 2013.

[9] 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.

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

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

[12] E. Veach and L. Guibas, "Optimally combining sampling techniques for Monte Carlo rendering," *Proceedings of the 22nd Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH)*, pp. 419–428, 1995.