STICKY PROPOSAL DENSITIES FOR ADAPTIVE MCMC METHODS

L. Martino**, R. Casarin*$^\diamond$*, D. Luengo*$^\sharp$

* Institute of Mathematical Sciences and Computing, Universidade de São Paulo, São Carlos (Brazil)
$^\diamond$ Dep. of Economics, University Ca' Foscari of Venice, Venice (Italy)
$^\sharp$ Dep. of Signal Theory and Communications, Universidad Politécnica de Madrid, Madrid (Spain)

ABSTRACT

Monte Carlo (MC) methods are commonly used in Bayesian signal processing to address complex inference problems. The performance of any MC scheme depends on the similarity between the proposal (chosen by the user) and the target (which depends on the problem). In order to address this issue, many adaptive MC approaches have been developed to construct the proposal density iteratively. In this paper, we focus on adaptive Markov chain MC (MCMC) algorithms, introducing a novel class of adaptive proposal functions that progressively "stick" to the target. This proposed class of sticky MCMC methods converge very fast to the target, thus being able to generate virtually independent samples after a few iterations. Numerical simulations illustrate the excellent performance of the sticky proposals when compared to other adaptive and non-adaptive schemes.

Index Terms**—** Statistical inference, Bayesian signal processing, Monte Carlo methods, adaptive MCMC

1. INTRODUCTION

A common problem in statistical signal processing is inferring some parameters of interest given a set of observations or measurements. Bayesian inference addresses this problem by defining a prior probability density function (pdf), that encompasses all the information available before performing the experiment, and a likelihood (i.e., an input-output model), and constructing the posterior pdf through Bayes theorem. The optimal estimator of the parameters is then typically an integral function of the posterior pdf. In most practical applications this integral cannot be solved analytically, and Monte Carlo (MC) methods are commonly used to approximate it [1, 2, 3].

MC methods can be divided into two large families [2, 4]: importance sampling (IS) and Markov chain Monte Carlo (MCMC). In both families, samples are initially drawn from a density chosen by the user (the proposal pdf), and then either weighted appropriately (in IS) or checked for acceptance (in MCMC) to guarantee that the final set of weighted samples is distributed according to the target (which depends on the problem). However, regardless of the family, the performance of any MC algorithm is directly related to the similarity between the proposal and target densities. Consequently, many adaptive IS [5, 6, 7, 8, 9] and adaptive MCMC [10, 11, 12, 13] approaches have been developed to construct the proposal density iteratively. Nevertheless, several issues still remain: mixing for multi-modal targets with narrow peaks, applicability to a large class of practical densities, scalability to high-dimensional problems, etc.

In this paper, we address the first two issues, developing a universal method that allows us to sample virtually from any univariate target density. Furthermore, the proposed algorithm can be easily applied within some other scheme (like the Gibbs sampler) to sample efficiently from multivariate pdfs. Starting from a set of support points (either selected by the user or chosen randomly), our aim is building a non-parametric approximation of the target iteratively using some simple and efficient construction. This goal is achieved by incorporating previously drawn samples to the set of support points according to some suitable rule. As a result, the proposed approach builds a sequence of proposal functions that progressively "stick" to the target. The novel class of sticky MCMC methods includes the recently proposed IA^{2}RMS scheme as a particular case [14, 15], which in turn builds upon the well-known ARMS algorithm [16]. Sticky proposals converge very fast to the target, thus allowing us to generate virtually independent samples after a few iterations and improving the performance of other competing (both adaptive and non-adaptive) techniques.

The paper is structured as follows. Firstly, Section 2 states the problem and introduces the notation. Then, adaptive MC algorithms are briefly reviewed in Secion 3, and the novel adaptive sticky MCMC approach proposed is described in detail in Section 4. Finally, numerical simulations are provided in Section 5, and the conclusions close the paper in Section 6.

This work has been supported by Grant 2014/23160-6 of the São Paulo Research Foundation (FAPESP); the European Union's Seventh Framework Program (FP7/2007-2013) under grant agreements SYRTO-SSH-2012-320270 and 630677; the Italian Ministry of Education, University and Research (MIUR) PRIN 2010-11 grant MISURA; the Spanish government through the DISSECT (TEC2012-38058-C03-01) and MIMOD-PLC (TEC2015-64835-C3-3-R) projects; and the BBVA Foundation's MG-FIAR project ("I Convocatoria de Ayudas Fundación BBVA a Investigadores, Innovadores y Creadores Culturales").

2. PROBLEM STATEMENT

Bayesian signal processing usually requires computing some moment of a posterior probability density function (pdf),

$$I = \int_{\mathcal{X}} f(\mathbf{x}) p(\mathbf{x}|\mathbf{y}) d\mathbf{x}, \tag{1}$$

where $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^{D_x}$ is the set of parameters to be inferred, $f(\mathbf{x})$ can be any square-integrable function, $\mathbf{y} \in \mathcal{Y} \subseteq \mathbb{R}^{N \times D_y}$ is the set of N observations or measurements, and the *posterior* pdf, $p(\mathbf{x}|\mathbf{y})$, is obtained through Bayes theorem:

$$p(\mathbf{x}|\mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{x}) p(\mathbf{x})}{p(\mathbf{y})}, \tag{2}$$

where $p(\mathbf{x})$ is the *prior* density, $p(\mathbf{y}|\mathbf{x})$ is the *likelihood*, and $p(\mathbf{y}) = \int_{\mathcal{X}} p(\mathbf{y}|\mathbf{x}) p(\mathbf{x}) d\mathbf{x}$ is the *evidence*. For instance, (1) corresponds to the well-known and widely used minimum mean squared error (MMSE) estimator when $f(\mathbf{x}) = \mathbf{x}$. Unfortunately, an analytical solution of (1) cannot be found in most practical applications. In these cases, Monte Carlo (MC) methods are commonly used to approximate I.

3. ADAPTIVE MONTE CARLO

MC methods are simulation-based techniques that provide an approximate solution to Eq. (1). For the sake of simplicity, let us denote the posterior or target pdf as $\bar{\pi}(\mathbf{x}) = \frac{1}{Z(\mathbf{y})} \pi(\mathbf{x})$, where $Z(\mathbf{y}) = p(\mathbf{y})$. If we are able to draw samples $\mathbf{x}_t \sim \bar{\pi}(\mathbf{x})$ for $t = 1, \ldots, T$, then I can be approximated as

$$\hat{I} = \frac{1}{T} \sum_{t=1}^{T} f(\mathbf{x}_t). \tag{3}$$

However, sampling directly from the target is usually unfeasible. In this cases, samples are drawn from some simpler proposal pdf, i.e., $\mathbf{x}_t \sim q(\mathbf{x})$ for $t = 1, \ldots, T$, and then (1) is approximated as

$$\hat{I} = \frac{1}{T} \sum_{t=1}^{T} w(\mathbf{x}_t) f(\mathbf{x}_t), \tag{4}$$

where $w(\mathbf{x}_t)$ denotes some properly selected weights. Two large families of MC methods exist [2, 4]:

- Importance sampling (IS) algorithms, that weight the samples drawn from $q(\mathbf{x})$ as $w(\mathbf{x}_t) = \frac{\bar{\pi}(\mathbf{x}_t)}{q(\mathbf{x}_t)}$ in order to obtain unbiased and consistent estimators.[1]
- Markov chain Monte Carlo (MCMC) schemes, that construct a Markov chain whose stationary distribution is the target, $\bar{\pi}(\mathbf{x})$, by drawing samples from $q(\mathbf{x})$ and accepting or rejecting them. In this case, $w(\mathbf{x}_t) \in \mathbb{Z}^+$ is simply the number of times that each sample is used to construct the final estimator.

In both families, as well as in the many hybrid approaches described in the literature that combine IS and MCMC (see, e.g., [18]), the performance of the method is directly related to the similarity between the proposal and the target densities. For this reason, many adaptive IS and adaptive MCMC methods have been proposed. In all cases, the goal is exploiting the information from the previously drawn samples to construct an improved sequence of proposal densities: $q_1(\mathbf{x}), q_2(\mathbf{x}|\mathbf{x}_1), \ldots, q_T(\mathbf{x}|\mathbf{x}_{1:T-1})$.

Focusing on adaptive MCMC methods, we can distinguish between *parametric* approaches, which update some (or all) of the parameters of some pre-defined proposal density, and *non-parametric* approaches, that build a proposal incrementally by using some suitable procedure. Regarding the first class of methods, they include the seminal work by Haario, where he used a random walk Gaussian proposal and adapted the covariance matrix using an empirical estimator [10]. This work was recently extended to a mixture of Gaussians, where all the parameters were updated using simple empirical estimators [13]. Regarding the second class, the best known example is the adaptive rejection Metropolis sampling (ARMS) algorithm [16], which constructs a piecewise linear proposal in the log-domain using a set of support points that is built iteratively from previously rejected samples. Although ARMS provides a good performance in many applications, there is a flaw in its adaptive mechanism that prevents the full convergence of the proposal to the target, as shown in [14, 15], where an improved method was also proposed: independent doubly adaptive rejection Metropolis sampling (IA^{2}RMS). The novel class of methods described in the following section generalizes both IA^{2}RMS and ARMS.

4. ADAPTIVE INDEPENDENT STICKY MCMC

Let $\bar{\pi}(x) \propto \pi(x)$, with $x \in \mathcal{X} \subseteq \mathbb{R}$, be a bounded target pdf. Let $\mathcal{S}_0 = \{s_0, \ldots, s_{m_0}\}$ be the initial set of support points (either selected by the user or chosen randomly), with $m_0 = |\mathcal{S}_0|$ denoting the cardinality of $\mathcal{S}_0$, and $\mathcal{S}_t = \{s_1, \ldots, s_{m_t}\}$ be the set of support points at the t-th iteration. An adaptive sticky MCMC method is composed of the following three stages:

1. *Construction of the proposal:* Given $\mathcal{S}_t$, build a proposal function $q_t(x|\mathcal{S}_t)$ using some appropriate non-parametric procedure.
2. *MCMC stage:* Apply some MCMC algorithm to obtain x_{t+1} given x_t, employing $q_t(x|\mathcal{S}_t)$ as the proposal pdf.
3. *Update of the support set:* Perform some suitable statistical test to decide whether a new point is added to $\mathcal{S}_t$ in order to obtain $\mathcal{S}_{t+1}$ or not.

[1] Let us remark that other weights are possible in multiple importance sampling (MIS) schemes, where multiple proposals $(q_1, \ldots, q_M)$ are used instead of a single one. See [17] for further details.

In the following sections we describe in detail an adaptive sticky MCMC algorithm.

4.1. Adaptive independent sticky Metropolis (AISM)

The simplest method belonging to the class of sticky MCMC is the *adaptive independent sticky Metropolis* (AISM) technique. At each iteration, AISM builds a non-parametric proposal function using one of the approaches described in Section 4.2, performs a single Metropolis-Hastings (MH) step using this proposal, and decides whether the set of support points should be updated according to a suitable rule that fulfills the conditions enumerated in Section 4.3. Table 1 summarizes the AISM algorithm. The key novelty of AISM is the statistical control applied in Step 3 to decide whether the set $\mathcal{S}_t$ should be updated or not. The point z, rejected in the previous MH test, is added to $\mathcal{S}_t$ with probability

$$P_a(z) = \eta_t(z, d_t(z)), \qquad (5)$$

where $d_t(z)$ denotes any valid distance function between the target and proposal functions (like the L_1 distance defined in Eq. (6) or the distance in Eq. (7) that is used in the simulations) evaluated at z, and $\eta_t(z, d_t(z))$ is some suitable acceptance rule, as detailed in Section 4.3. The rationale behind this test is to use information from the target density in order to include in the support set only those points where the proposal pdf differs substantially from the target. Note that, since z is always different from the current state (i.e., $z \neq x_t$ for all t), the proposal pdf is *independent* from the current state according to Holden's definition [19], and thus the theoretical analysis of AISM is greatly simplified [20].

4.2. Construction of the sticky proposals

There are many valid alternatives for the construction of a suitable *sticky proposal* (SP) pdf. Given the set of support points $\mathcal{S}_t$, there are three properties that an SP must satisfy:

1. The proposal function always has to be positive, i.e., $q_t(x|\mathcal{S}_t) > 0$ for all $x \in \mathcal{X}$ and any valid $\mathcal{S}_t$.
2. The L_1 distance between $\pi(x)$ and $q_t(x)$ must tend to zero as the number of support points increases, i.e.,

$$\|\pi(x) - q_t(x|\mathcal{S}_t)\|_1 = \int_{\mathcal{X}} |\pi(x) - q_t(x|\mathcal{S}_t)| dx \to 0 \qquad (6)$$

   as $m_t \to \infty$ for all $x \in \mathcal{X}$.

3. Samples can be drawn directly and easily from $q_t(x|\mathcal{S}_t)$ using some exact sampling procedure.

All of the approaches described in [14, 15], which build a piecewise constant or linear proposal pdf (either in the pdf's domain or in the log-domain) directly from the support points with a suitable construction for the tails, fulfill these three properties and can thus be used here. Other alternative procedures can also be found in the literature [16, 21, 22].

Table 1. Adaptive Independent Sticky Metropolis (AISM)

```
For $t = 0, \ldots, T-1$:
    1. Construction of the proposal: Build a proposal function $q_t(x|\mathcal{S}_t)$ via a suitable interpolation procedure using the set of support points $\mathcal{S}_t$ (see Section 4.2).
    2. MH step:
        2.1 Draw $x' \sim q_t(x|\mathcal{S}_t)$.
        2.2 Set $x_{t+1} = x'$ and $z = x_t$ with probability
                $\alpha = \min \left[1, \frac{\pi(x')q_t(x_{t+1}|\mathcal{S}_t)}{\pi(x_{t+1})q_t(x'|\mathcal{S}_t)}\right].$
            Otherwise (i.e., with probability $1-\alpha$), set $x_{t+1} = x_t$ and $z = x'$.
    3. Test to update $\mathcal{S}_t$: Let $\eta_t(z, d_t(z))$ be a suitable update rule (see Section 4.3), with $d_t(z)$ denoting some valid distance function. Set
            $\mathcal{S}_{t+1} = \begin{cases} \mathcal{S}_t \cup \{z\}, & \text{with prob. } P_a(z) = \eta_t(z, d_t(z)), \\ \mathcal{S}_t, & \text{with prob. } 1 - P_a(z), \end{cases}$
```

4.3. Update of the set of support points

A suitable choice of the function, $\eta_t(z, d)$ with $d = d_t(z)$, is required in AISM. Any valid test function $\eta_t(z, d)$ must fulfill the following properties:

1. $\eta_t(z, d) : \mathcal{X} \times \mathbb{R}^+ \to [0, 1]$.
2. $\frac{\partial \eta_t(z,d)}{\partial d} \geq 0$.
3. $\eta_t(z, 0) = 0$.
4. $\lim_{d \to \infty} \eta_t(z, d) = 1$.

Note that, for a given value of z, η_t satisfies all the properties of a continuous distribution function (cdf) associated to a positive random variable. Therefore, any pdf for positive random variables can be used to define a valid test function η_t through its corresponding cdf. Note also that η_t controls the trade-off between the performance and the computational cost of AISM, since $P_a(z) = \eta_t(z, d_t(z))$ is the probability of incorporating z to the set of support points. Indeed, the use of a large number of support points improves the performance (as the proposal becomes closer to the target) at the expense of a higher storage and computational cost. Hence, a good adaptive strategy should only include new points in regions where there is a large discrepancy between the proposal and the target functions. In the following section we provide one example of such test function.

5. NUMERICAL SIMULATIONS

In order to check the performance of AISM, we test it on a mixture model with heavy tailed components. More specifi-

Algorithm	Mean	SD	ACF(1)	ACF(10)	ACF(50)	m_T	$\underline{c}_T$	$\bar{c}_T$	Time
Mix 1									
Slice	19.5039	6.0238	0.8759	0.8230	0.6244	-	0.9934	1.0089	0.8257
ARMS	20.2803	10.3538	0.8593	0.8112	0.6848	42.5690	0.7417	1.7490	1.0000
AISM	19.1416	2.7723	0.1182	0.0951	0.0785	112.7360	0.9933	1.0085	0.4679
Mix 2 ($\kappa = 0.1$)									
Slice	53.0797	14.5540	0.6835	0.3562	0.2666	-	0.9563	1.3878	0.6670
ARMS	61.1859	3.4341	0.0625	0.0219	0.0079	59.3500	0.7651	1.6203	1.0000
AISM	61.9253	1.6282	0.0283	0.0015	0.0005	121.1240	0.9575	1.1836	0.4344
Mix 2 ($\kappa = 0.4$)									
Slice	33.4459	4.6767	0.6933	0.5131	0.2230	-	0.9895	1.0113	0.4557
ARMS	33.9293	1.0835	0.1451	0.0375	0.0047	57.7728	0.9622	1.0394	1.0000
AISM	33.8768	0.7482	0.0247	0.0013	0.0007	131.7785	0.9896	1.0112	0.5212

Table 2. Results of the slice sampler, ARMS and AISM for Mix1 (upper portion) and Mix2(κ) (lower portion). Each row shows the mean, the standard deviation (SD), the autocorrelation function (ACF(k)) at lags $k \in \{1, 10, 50\}$, the final number of support points (m_T), the final estimates of the normalizing constant ($\underline{c}_T$, $\bar{c}_T$), and the normalized computing time.

cally, we consider the following mixtures of generalized exponential power (GEP) distributions:

1. Mixture of heavy and normal-tailed *symmetric* distributions (Mix 1):

$$\bar{\pi}(x) = 0.6\,\mathcal{GEP}(0, 1, 1/2, 1) + 0.4\,\mathcal{GEP}(50, 1, 2, 1).$$

2. Mixture of heavy and normal-tailed *asymmetric* distributions (Mix 2 (κ) with $\kappa \in \{0.1, 0.4\}$):

$$\bar{\pi}(x) = 0.4\,\mathcal{GEP}(0, 1, 1/2, 2) + 0.6\,\mathcal{GEP}(50, 1, 1/2, \kappa).$$

The notation $\mathcal{GEP}(\mu, \sigma^2, \alpha, \kappa)$ indicates a GEP distribution with location, scale, shape and asymmetry parameters μ, σ, α and κ, respectively. The shape parameter α controls the tails of the density function and determines whether it is flat or peaked. The parameter κ is an inverse scale factor which controls the asymmetry of the distribution. When $\alpha = 2$ (and $\kappa = 1$) we have the Gaussian distribution, and when $\alpha = 1$ (and $\kappa = 1$) we have the Laplacian distribution. A smaller value of α corresponds to a heavy tailed distribution, and when $\alpha \to 0$ we have the Dirac mass centered at μ. The GEP distribution has the exponential power (EP) distribution, which has been successfully applied in many fields (see [23] for a review), as special case for $\kappa = 1$.

In the sequel we test the performance of AISM, using a piecewise linear proposal directly in the pdf's domain (see [14, 15] for further details), and a random test function:

$$\eta(z, d_t(z)) = d_t(z)^{\beta},$$
$$d_t(z) = 1 - \frac{\min\{\pi(z), q_t(z|\mathcal{S}_t)\}}{\max\{\pi(z), q_t(z|\mathcal{S}_t)\}}, \qquad (7)$$

with $\beta = 1$.[2] For the sake of comparison we use ARMS [16] and the slice sampler [24]. We generate $T = 5,000$ draws from each algorithm and compute (averaged over 2,000 independent runs without removing any burn-in samples) the mean, the autocorrelation function, and the final number of support points. We start ARMS and AISM with the same initial value and set of support points, $\mathcal{S}_0 = \{-1, 1, 20\}$. The results for two different experiments are shown in Table 2. In general, the performance of the algorithms in terms of estimated mean is similar (note that the true mean is 20 in the first case, 62 in the second, and 34 in the third). However, AISM substantially improves the performance of the other two methods in terms of the standard deviation of the estimates and the auto-correlation of the samples generated. Furthermore, AISM is also more efficient in terms of normalized computation time than the other two algorithms.

6. CONCLUSIONS AND FUTURE LINES

In this paper, we have introduced a novel class of adaptive MCMC methods which are based on iteratively building a non-parametric proposal function that progressively "sticks" to the target pdf. The AISM algorithm described in the paper is able to provide virtually independent samples and an improved performance w.r.t. other state-of-the-art adaptive (ARMS) and non-adaptive (slice sampler) techniques in one-dimensional problems. Future lines include exploring the use of sticky proposals with more sophisticated MCMC schemes, like the multiple try Metropolis (MTM) [25, 26, 27], and testing the proposed approach in higher dimensional problems by embedding it within another MCMC technique, like Gibbs sampling or the hit-and-run algorithm.

[2] Note that $\beta \in (0, +\infty)$. When $\beta < 1$, the incorporation of new support points to $\mathcal{S}_t$ is facilitated, whereas it is discouraged with $\beta > 1$.

7. REFERENCES

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

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

[3] A. Doucet and X. Wang, "Monte Carlo methods for signal processing: a review in the statistical signal processing context," *IEEE Signal Processing Magazine*, vol. 22, no. 6, pp. 152–170, 2005.

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

[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] 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.

[8] L. Martino, V. Elvira, D. Luengo, and J. Corander, "An adaptive population importance sampler," *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 8088–8092, 2014.

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

[10] H. Haario, E. Saksman, and J. Tamminen, "An adaptive Metropolis algorithm," *Bernoulli*, vol. 7, no. 2, pp. 223–242, April 2001.

[11] C. Andrieu and J. Thoms, "A tutorial on adaptive MCMC," *Statistics and Computing*, vol. 18, no. 4, pp. 343–373, 2008.

[12] G. O. Roberts and J. S. Rosenthal, "Examples of adaptive MCMC," *Journal of Computational and Graphical Statistics*, vol. 18, no. 2, pp. 349–367, 2009.

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

[14] L. Martino, J. Read, and D. Luengo, "Independent doubly adaptive rejection Metropolis sampling," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 7998–8002.

[15] L. Martino, J. Read, and D. Luengo, "Independent doubly adaptive rejection Metropolis sampling within Gibbs sampling," *IEEE Transactions on Signal Processing*, vol. 63, no. 12, pp. 3123–3138, 2015.

[16] W. R. Gilks, N. G. Best, and K. K. C. Tan, "Adaptive Rejection Metropolis Sampling within Gibbs Sampling," *Applied Statistics*, vol. 44, no. 4, pp. 455–472, 1995.

[17] V. Elvira, L. Martino, D. Luengo, and M. F. Bugallo, "Generalized multiple importance sampling," *arXiv:1511.03095*, 2015.

[18] L. Martino, V. Elvira, D. Luengo, and J. Corander, "Layered adaptive importance sampling," *Statistics and Computing (accepted; in press)*, 2016.

[19] L. Holden, R. Hauge, and M. Holden, "Adaptive independent Metropolis-Hastings," *The Annals of Applied Probability*, vol. 19, no. 1, pp. 395–413, 2009.

[20] L. Martino, R. Casarin, F. Leisen, and D. Luengo, "Adaptive sticky generalized Metropolis," *arXiv:1308.3779*, 2016.

[21] W. R. Gilks and P. Wild, "Adaptive Rejection Sampling for Gibbs Sampling," *Applied Statistics*, vol. 41, no. 2, pp. 337–348, 1992.

[22] R. Meyer, B. Cai, and F. Perron, "Adaptive rejection Metropolis sampling using Lagrange interpolation polynomials of degree 2," *Computational Statistics and Data Analysis*, vol. 52, no. 7, pp. 3408–3423, March 2008.

[23] S. Kotz, T. J. Kozubowoski, and K. Podgórski, *The Laplace Distribution and Generalizations: A Revisit with Applications to Communications, Economics, Engineering and Finance*, Birkhöuser, 2001.

[24] R. M. Neal, "Slice sampling," *Annals of Statistics*, vol. 31, no. 3, pp. 705–767, 2003.

[25] R. Casarin, R. Craiu, and F. Leisen, "Interacting multiple try algorithms with different proposal distributions," *Statistics and Computing*, vol. 23, no. 2, pp. 185–200, 2013.

[26] L. Martino and J. Read, "On the flexibility of the design of multiple try Metropolis schemes," *Computational Statistics*, vol. 28, no. 6, pp. 2797–2823, 2013.

[27] L. Martino and F. Louzada, "Issues in the Multiple Try Metropolis mixing," *Computational Statistics (accepted; in press)*, 2016.