

INDEPENDENT DOUBLY ADAPTIVE REJECTION METROPOLIS SAMPLING

Luca Martino , Jesse Read , David Luengo

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

Adaptive Rejection Metropolis Sampling (ARMS) is a well-known MCMC scheme for generating samples from one-dimensional target distributions. ARMS is widely used within Gibbs sampling, where automatic and fast samplers are often needed to draw from univariate full-conditional densities. In this work, we propose an alternative adaptive algorithm (IA²RMS) that overcomes the main drawback of ARMS (an uncomplete adaptation of the proposal in some cases), speeding up the convergence of the chain to the target. Numerical results show that IA²RMS outperforms the standard ARMS, providing a correlation among samples close to zero.

1. INTRODUCTION

Markov chain Monte Carlo (MCMC) methods generate samples from a target probability density function (PDF) by drawing from a simpler proposal PDF [1, 2]. The two best known MCMC approaches are the Metropolis-Hastings (MH) algorithm and the Gibbs sampler [3]. Gibbs sampling produces samples from multi-dimensional target densities, drawing each component of the generated samples from the corresponding univariate full-conditional density. The key point for its successful application is being able to draw efficiently from these univariate PDFs. The best scenario for Gibbs sampling occurs when exact samplers for each full-conditional are available. Otherwise, sampling techniques like rejection sampling (RS) or some variant of the MH algorithm are used *within* the Gibbs sampler to draw from complicated full-conditionals. In the first case, samples generated from the RS algorithm are independent, but the acceptance rate can be very low. In the second case, we have an MCMC-inside-another-MCMC approach. Therefore, the typical problems of the *external*-MCMC (long “burn-in” period, large correlation, etc.) could raise dramatically if the *internal*-MCMC is not extremely efficient.

This work is funded by the Spanish government’s projects COMONSENS (CSD2008-00010), COMPREHENSION (TEC2012-38883-C02-01), ALCIT (TEC2012-38800-C03-01), DISSECT (TEC2012-38058-C03-01).

Automatic and self-tuning samplers, such as *adaptive rejection sampling* (ARS) [4, 5], have been developed to draw efficiently from univariate target densities. The samples generated by ARS are independent and the proposal always converges to the true shape of the target, but ARS can only be applied for log-concave (i.e., unimodal) targets. In order to overcome this limitation, *Adaptive Rejection Metropolis Sampling* (ARMS) [6, 7] combines the ARS and MH approaches. ARMS is a universal sampler that builds a self-tuning proposal (i.e., a proposal automatically constructed and adapted to the target), but the generated samples are correlated and the adaptation mechanism of ARMS is uncomplete: in some cases the proposal does not converge to the target in certain areas or even in the whole domain [8, 9].

In this work, we present an enhancement of ARMS which ensures that the sequence of proposals converges to the target, while maintaining the computational cost bounded (exactly as in ARS and ARMS) with the addition of a simple control test. We call the novel approach *independent A²RMS* (IA²RMS), since the proposal is independent of the current state and the A² emphasizes that we incorporate an additional adaptive control. The new strategy allows us to decouple completely the adaptation mechanism from the proposal construction (unlike ARMS, whose performance depends critically on the proposal building approach), thus allowing us to introduce several examples of simpler construction procedures. Numerical simulations on a Gaussian mixture example show the effectiveness of the proposed approach.

2. PROBLEM STATEMENT

Bayesian inference often requires drawing samples from complicated multivariate posterior PDFs, $p(\mathbf{x}|\mathbf{y})$ with $\mathbf{x} \in \mathcal{D} \subseteq \mathbb{R}^D$. A common approach, when direct sampling from $p(\mathbf{x}|\mathbf{y})$ is unfeasible, is using a Gibbs sampler [3]. At the i -th iteration, a Gibbs sampler obtains the d -th component ($d \in \{1, \dots, D\}$) of \mathbf{x} , x_d , drawing from the full conditional of x_d given all the information available [10, 11], i.e.,

$$x_d^{(i)} \sim p(x_d | \mathbf{x}_{1:d-1}^{(i)}, \mathbf{x}_{d:D}^{(i-1)}), \quad (1)$$

with the initial vector drawn from the prior, i.e., $\mathbf{x}^{(0)} \sim p_0(\mathbf{x})$.

However, even sampling from (1) can often be complicated. In these cases, a common approach is using another Monte Carlo technique (e.g., RS or the MH algorithm) within the Gibbs sampler, drawing candidates from a simpler proposal, $\pi(x)$. The performance of this approach depends greatly on the choice of $\pi(x)$. Thus, adaptive strategies, where the proposal is updated using the previously generated samples, are usually preferred. For the sake of simplicity, in the sequel we denote the univariate target PDF (i.e., the full-conditional proposal in Eq. (1)) as $p(x)$, and the sequence of proposals as $\pi_t(x)$ for $t = 1, \dots, T$. Our aim is designing a sequence of self-tuning proposals such that $\pi_t(x) \rightarrow p(x)$ when $t \rightarrow \infty$ as fast as possible.

3. ADAPTIVE REJECTION METROPOLIS SAMPLING (ARMS)

Adaptive rejection Metropolis sampling (ARMS) combines the adaptive rejection sampling (ARS) and the Metropolis-Hastings (MH) [2, 3] techniques. It first performs an RS test, and the rejected samples are then used to improve the proposal PDF, exactly as in the standard ARS. ARMS is summarized in Table 1. The proposal construction approach is critical for the good performance of ARMS. Let us consider a set of *support points* at time t , $\mathcal{S}_t = \{s_1, s_2, \dots, s_{m_t}\} \subset \mathcal{D}$, s.t. $s_1 < \dots < s_{m_t}$; the intervals $\mathcal{I}_0 = (-\infty, s_1]$, $\mathcal{I}_j = (s_j, s_{j+1}]$ for $j = 1, \dots, m_t - 1$ and $\mathcal{I}_{m_t} = (s_{m_t}, +\infty)$; and $V(x) = \log(p(x))$. Moreover, let us denote as $L_{j,j+1}(x)$ the line passing through $(s_j, V(s_j))$ and $(s_{j+1}, V(s_{j+1}))$ for $j = 1, \dots, m_t - 1$, with $L_{-1,0}(x) = L_{0,1}(x) = L_{1,2}(x)$ and $L_{m_t,m_t+1}(x) = L_{m_t+1,m_t+2}(x) = L_{m_t-1,m_t}(x)$. Then, in [6] a piecewise-linear function $W_t(x)$ is constructed as

$$W_t(x) = \max [L_{j,j+1}(x), \min [L_{j-1,j}(x), L_{j+1,j+2}(x)]] , \quad (2)$$

for $x \in \mathcal{I}_j = (s_j, s_{j+1}]$ and $j = 0, \dots, m_t$. Hence, the proposal PDF, $\tilde{\pi}_t(x) \propto \pi_t(x) = \exp(W_t(x))$, is formed by exponential pieces. Fig. 1(a) illustrates an example of this construction. More sophisticated approaches to build $W_t(x)$ (e.g., using quadratic segments when possible [12]) have been proposed. However, none of these schemes solves the structural problem of ARMS that is briefly described next.

Note that, when a sample is rejected by ARMS in the RS test (this can only happen when $\pi_t(x') > p(x')$), this new point is always added to the set \mathcal{S}_{t+1} and used to update the proposal, $\pi_{t+1}(x)$. However, when a sample is initially accepted by the RS test (as it always happens when $\pi_t(x') \leq p(x')$), the proposal is never updated, regardless of whether that point is finally accepted in the MH test or not. This causes the proposal adaptation procedure to be uncomplete in some cases (see [8, 9] for a detailed discussion and an example). For this reason, the performance of ARMS depends critically on the procedure used to build the proposal, which is not truly independent of the adaptation mechanism.¹

¹Indeed, in order to obtain satisfactory results with ARMS, the procedure

Table 1. ARMS algorithm.

<p>Initialization:</p> <ol style="list-style-type: none"> 1. Set $k = 0$ (chain's iteration), $t = 0$, choose an initial state x_0 and support set $\mathcal{S}_0 = \{s_1, \dots, s_{m_0}\}$. <p>Iterations (while $k < N$):</p> <ol style="list-style-type: none"> 2. Build a proposal, $\pi_t(x \mathcal{S}_t)$, given a set of support points $\mathcal{S}_t = \{s_1, \dots, s_{m_t}\}$, according to Eq. (2). 3. Draw $x' \sim \tilde{\pi}_t(x \mathcal{S}_t) \propto \pi_t(x \mathcal{S}_t)$ and $u' \sim \mathcal{U}([0, 1])$. 4. If $u' > \frac{p(x')}{\pi_t(x' \mathcal{S}_t)}$, reject x', set $\mathcal{S}_{t+1} = \mathcal{S}_t \cup \{x'\}$ and update $t = t + 1$. Go back to step 2. 5. Otherwise, set $x_{k+1} = x'$ with probability $\alpha = \min \left[1, \frac{p(x') \min [p(x_k), \pi_t(x_k \mathcal{S}_t)]}{p(x_k) \min [p(x'), \pi_t(x' \mathcal{S}_t)]} \right],$ or $x_{k+1} = x_k$ and $y = x'$ with probability $1 - \alpha$. Set $\mathcal{S}_{t+1} = \mathcal{S}_t$, $t = t + 1$ and $k = k + 1$. 6. If $k < N$, go back to step 2.

4. INDEPENDENT DOUBLY ADAPTIVE REJECTION METROPOLIS SAMPLING (IA²RMS)

In this section, we describe a simple and extremely efficient strategy which allows us to solve the two aforementioned problems of ARMS. The novel scheme ensures the convergence of the chain to the target distribution and keeps, at the same time, the computational cost bounded. Furthermore, it allows us to completely decouple the adaptation mechanism from the proposal construction, thus allowing us to consider simpler alternatives for the latter. We call it *independent doubly adaptive rejection Metropolis sampling* (IA²RMS), where the A² emphasizes that we incorporate an additional adaptive step to improve the proposal PDF w.r.t. ARMS. The IA²RMS algorithm is summarized in Table 2. The key point is the introduction of this new control step (step 5.2), which allows us to add samples (in a controlled way) inside regions of the domain where $\pi_t(x|\mathcal{S}_t) < p(x)$.

4.1. Convergence and Computational cost

The new control test is performed using an auxiliary variable, y , which is always different from the new state, x_{k+1} . This construction leads to a proposal, $\pi_t(x|\mathcal{S}_t)$, which is independent of the current state of the chain, x_k . Therefore, the

used to build $W_t(x)$ must fulfill the two requirements described in [9], thus leading to unnecessarily complex proposal construction schemes.

Table 2. IA²RMS algorithm.

<p>Initialization:</p> <ol style="list-style-type: none"> 1. Set $k = 0$ (chain's iteration), $t = 0$, choose an initial state x_0 and support set $\mathcal{S}_0 = \{s_1, \dots, s_{m_0}\}$. <p>Iterations (while $k < N$):</p> <ol style="list-style-type: none"> 2. Build a proposal, $\pi_t(x \mathcal{S}_t)$, given the set $\mathcal{S}_t = \{s_1, \dots, s_{m_t}\}$, using a convenient procedure. 3. Draw $x' \sim \tilde{\pi}_t(x \mathcal{S}_t) \propto \pi_t(x \mathcal{S}_t)$ and $u' \sim \mathcal{U}([0, 1])$. 4. If $u' > \frac{p(x')}{\pi_t(x' \mathcal{S}_t)}$, then reject x', set $\mathcal{S}_{t+1} = \mathcal{S}_t \cup \{x'\}$, update $t = t + 1$ and go back to step 2. 5. Otherwise, i.e., if $u' \leq \frac{p(x')}{\pi_t(x' \mathcal{S}_t)}$, then: <ol style="list-style-type: none"> 5.1 Set $x_{k+1} = x'$ and $y = x_k$ with probability $\alpha = \min \left[1, \frac{p(x') \min[p(x_k), \pi_t(x_k \mathcal{S}_t)]}{p(x_k) \min[p(x'), \pi_t(x' \mathcal{S}_t)]} \right],$ or $x_{k+1} = x_k$ and $y = x'$ with prob. $1 - \alpha$. 5.2 Draw $u_2 \sim \mathcal{U}([0, 1])$. If $u_2 > \frac{\pi_t(y \mathcal{S}_t)}{p(y)},$ set $\mathcal{S}_{t+1} = \mathcal{S}_t \cup \{y\}$. Otherwise, $\mathcal{S}_{t+1} = \mathcal{S}_t$. 5.3 Update $t = t + 1$ and $k = k + 1$. 6. If $k < N$, go back to step 2.

convergence of the Markov chain to a bounded target density is ensured by a theoretical result from [13] (see also [1, Chapter 8]). Indeed, IA²RMS satisfies the strong Doeblin's condition if the target is bounded: in this case the proposal is bounded and the construction of the tails can always be chosen adequately (see Section 5 for a brief discussion and [14] for a rigorous proof concerning an alternative adaptive method). Moreover, since $\{\pi_t(x)\}_{t=0}^{+\infty}$ always approaches the target $p(x)$, IA²RMS satisfies the *diminishing adaptation* condition [1, Chapter 8], i.e., the discrepancy between the proposals $\pi_t(x)$ and $\pi_{t+1}(x)$ becomes progressively smaller as $t \rightarrow \infty$. Consequently, the probability of updating the proposal PDF eventually becomes zero, thus preventing the addition of new support points.

The coding and implementation complexity of IA²RMS is virtually identical to ARMS, since all the quantities involved in the ratio of step 5.2 have been previously calculated in steps 4 and 5.1. Thus, no additional evaluation of the proposal and target PDFs is required. The total number of support points

increases w.r.t. ARMS, although it always remains within the same order of magnitude, as shown in the simulations. Indeed, it is important to emphasize that the number of support points does not diverge: it is kept bounded thanks to the two control tests, exactly as in ARS and ARMS. This is due to the fact that both $\frac{p(x)}{\pi_t(x|\mathcal{S}_t)}$ and $\frac{\pi_t(x|\mathcal{S}_t)}{p(x)}$ measure the point-wise discrepancy between the proposal and the target. As the proposal approaches the target, the two ratios tend to one, and the probability of adding a new support point vanishes quickly to zero, as in ARS and ARMS.

5. ALTERNATIVE PROPOSAL CONSTRUCTION

Since IA²RMS improves the adaptive structure of ARMS, simpler procedures can be used to build the function $W_t(x)$, thus reducing the overall computational cost and the coding effort. A first possibility is defining $W_t(x)$ inside the i -th interval simply as the straight line $L_{i,i+1}(x)$ going through $(s_i, V(s_i))$ and $(s_{i+1}, V(s_{i+1}))$ for $1 \leq i \leq m_t - 1$, and extending the straight lines corresponding to \mathcal{I}_1 and \mathcal{I}_{m_t-1} towards $\pm\infty$ for the first and last intervals. Mathematically,

$$W_t(x) = L_{i,i+1}(x), \quad x \in \mathcal{I}_i = (s_i, s_{i+1}], \quad (3)$$

for $1 \leq i \leq m_t - 1$, $W_t(x) = L_{1,2}(x)$ in $\mathcal{I}_0 = (-\infty, s_1]$ and $W_t(x) = L_{m_t-1,m_t}(x)$ in $\mathcal{I}_{m_t} = (s_{m_t}, \infty)$. This is illustrated in Fig. 1(b). Note that, although this procedure looks similar to the one used in ARMS (as described by Eq. (2)), it is much simpler, since it does not require the calculation of intersection points. Furthermore, an even simpler procedure to construct $W_t(x)$ can be devised from Eq. (3): using a piecewise constant approximation with two straight lines inside the first and last intervals. Mathematically,

$$W_t(x) = \max[V(s_i), V(s_{i+1})], \quad x \in \mathcal{I}_i = (s_i, s_{i+1}], \quad (4)$$

for $1 \leq i \leq m_t - 1$, $W_t(x) = L_{1,2}(x)$ in $\mathcal{I}_0 = (-\infty, s_1]$ and $W_t(x) = L_{m_t-1,m_t}(x)$ in $\mathcal{I}_{m_t} = (s_{m_t}, \infty)$. This construction leads to the simplest proposal possible: a collection of uniform PDFs with two exponential tails. Fig. 1(c) shows an example of the construction of the proposal using this approach. Finally, we can also apply the procedure described in [15] for adaptive trapezoidal Metropolis sampling (ATRAMS), even though the structure of this algorithm is completely different to ARMS. The proposal is constructed using straight lines $\tilde{L}_{i,i+1}(x)$ passing through $(s_i, p(s_i))$ and $(s_{i+1}, p(s_{i+1}))$ directly in the domain of the target PDF, $p(x)$. Mathematically,

$$\pi_t(x) = \tilde{L}_{i,i+1}(x), \quad x \in \mathcal{I}_i = (s_i, s_{i+1}], \quad 1 \leq i \leq m_t - 1, \quad (5)$$

and the tails are formed by two exponential pieces. Fig. 1(c) shows an example of a proposal using this approach. Note that, in all of the previous procedures, the construction of the tails can be modified in order to reduce the dependence on the initial set \mathcal{S}_0 or to adapt to specific classes of targets (e.g., corresponding to heavy-tailed distributions).

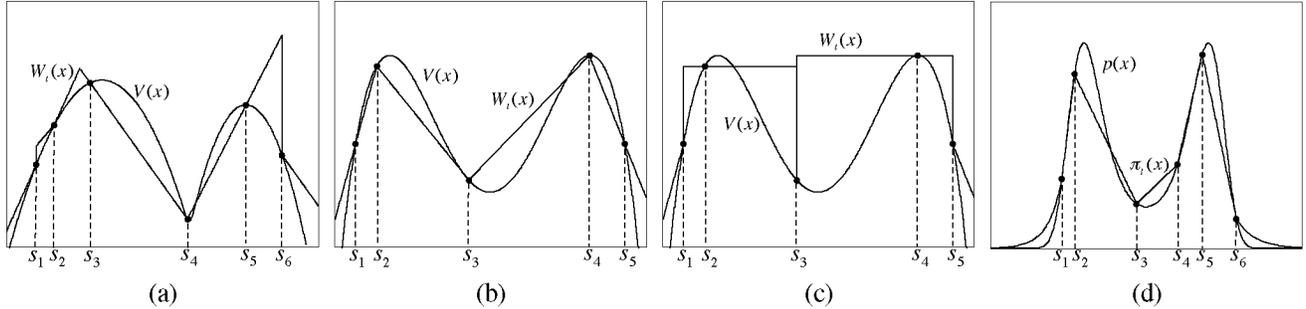


Fig. 1. Examples of proposal construction using: **(a)** the procedure proposed in ARMS [6]; **(b)** the procedure described in Eq. (3); **(c)** the procedure in Eq. (4); **(d)** the procedure in Eq. (5).

Table 3. Comparison of IA²RMS vs. ARMS adaptive structures. $\hat{\mu} \pm \hat{\sigma}$: estimated mean \pm standard deviation; $\hat{R}_{xx}[1]$: auto-correlation at lag 1; $\hat{D}_{\pi|p}(T)$: estimate of $D_{\pi|p}(t)$ at $t = T$; $|\mathcal{P}_T|$: average number of pieces in $\pi_t(x)$ at $t = T$.

Proposal $\pi_t(x)$	ARMS				IA ² RMS			
	$\hat{\mu} \pm \hat{\sigma}$	$\hat{R}_{xx}[1]$	$\hat{D}_{\pi p}(T)$	$ \mathcal{P}_T $	$\hat{\mu} \pm \hat{\sigma}$	$\hat{R}_{xx}[1]$	$\hat{D}_{\pi p}(T)$	$ \mathcal{P}_T $
Eq. (2)	1.648 ± 0.730	0.396	3.002	91.272	1.623 ± 0.124	0.004	0.061	123.844
Eq. (3)	1.754 ± 1.091	0.772	8.052	12.049	1.724 ± 0.219	0.020	0.253	85.644
Eq. (4)	1.594 ± 0.230	0.613	6.152	164.188	1.601 ± 0.095	0.002	0.201	317.536
Eq. (5)	1.567 ± 0.496	0.708	7.134	37.823	1.601 ± 0.131	0.005	0.058	92.133

6. NUMERICAL RESULTS

In this section, we compare the performance of IA²RMS and ARMS on a multimodal target PDF, $\tilde{p}(x)$, generated as a mixture of 3 Gaussian densities,²

$$\tilde{p}(x) = 0.3\mathcal{N}(x; -5, 1) + 0.3\mathcal{N}(x; 1, 1) + 0.4\mathcal{N}(x; 7, 1), \quad (6)$$

where $\mathcal{N}(x; \mu, \sigma^2)$ denotes a Gaussian PDF with mean μ and variance σ^2 . We consider the four alternative procedures previously described to build the proposal and estimate the mean of $\tilde{p}(x)$ (true value, $E\{\tilde{p}(x)\} = 1.6$) from the generated samples using both ARMS and IA²RMS. We also provide an estimation of the linear correlation among consecutive samples, the number of rejections, the number of pieces of the proposal, $|\mathcal{P}_T|$, and the distance, $D_{\pi|p}(t) = \int_{\mathcal{D}} |\pi_t(x) - p(x)| dx$, between the sequence of proposals ($t = 1, \dots, T$) and the target PDF [16].

In all cases, we consider $N = 5000$ iterations of the Markov chain, without removing any samples to reduce the burn-in period, and an initial support set $\mathcal{S}_0 = \{s_1 = -10, s_2 = a, s_3 = b, s_4 = 10\}$ formed by $m_0 = 4$ support points, where $a, b \sim \mathcal{U}([-10, 10])$ with $a < b$. Table 3 shows the results, averaged over 2000 runs, for ARMS and IA²RMS respectively. The standard ARMS method corresponds to the first row of Table 3. Note that IA²RMS always provides

better results than ARMS, regardless of the proposal construction scheme. This can be seen by the improvement in the estimated mean and, most notably, by the large decrease in the standard deviation of the estimation. We also notice that the correlation after N iterations is much lower for IA²RMS, due to the convergence of $\pi_t(x)$ to $p(x)$ almost everywhere, as evidenced also by the low value of $D_{\pi|p}(T)$. Indeed, IA²RMS with the proposal construction procedure of Eq. (5) provides much better results than the standard ARMS method (i.e., ARMS with the proposal of Eq. (2)) with the same computational cost (i.e., with the same number of pieces).

7. CONCLUSIONS

In this work, we have introduced a new adaptive technique (IA²RMS), which provides an automatic construction of a sequence of self-tuned proposals that always approaches the target (unlike ARMS), while keeping the computational cost bounded. As a consequence, the convergence of the chain is speeded up w.r.t. ARMS and the correlation vanishes quickly to zero. Furthermore, IA²RMS also allows us to reduce the complexity in the construction of the sequence of proposals. Thus, we have also proposed three simpler procedures to build the proposal densities. Numerical results show that IA²RMS outperforms ARMS in terms of estimation accuracy, correlation and convergence of the proposal to the target. Future work includes testing the performance of IA²RMS on more complex examples, where it is used within a Gibbs sampler.

²The comparison of IA²RMS with other MCMC approaches, as well as its performance for other types of targets (e.g., a heavy-tailed distribution) can be seen in [9].

8. REFERENCES

- [1] F. Liang, C. Liu, and R. Carroll, *Advanced Markov Chain Monte Carlo Methods: Learning from Past Samples*, Wiley Series in Computational Statistics, England, 2010.
- [2] J. S. Liu, *Monte Carlo Strategies in Scientific Computing*, Springer, 2004.
- [3] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*, Springer, 2004.
- [4] W. R. Gilks, “Derivative-free Adaptive Rejection Sampling for Gibbs Sampling,” *Bayesian Statistics*, vol. 4, pp. 641–649, 1992.
- [5] W. R. Gilks and P. Wild, “Adaptive Rejection Sampling for Gibbs Sampling,” *Applied Statistics*, vol. 41, no. 2, pp. 337–348, 1992.
- [6] 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.
- [7] W. R. Gilks, R.M. Neal, N. G. Best, and K. K. C. Tan, “Corrigendum: Adaptive Rejection Metropolis Sampling within Gibbs Sampling,” *Applied Statistics*, vol. 46, no. 4, pp. 541–542, 1997.
- [8] L. Martino and J. Míguez, “Generalized rejection sampling schemes and applications in signal processing,” *Signal Processing*, vol. 90, no. 11, pp. 2981–2995, November 2010.
- [9] L. Martino, J. Read, and D. Luengo, “Improved adaptive rejection Metropolis sampling algorithms,” Tech. Rep., Oct. 2012, arXiv:1205.5494v4.
- [10] J. Kotecha and Petar M. Djurić, “Gibbs sampling approach for generation of truncated multivariate Gaussian random variables,” *Proceedings of Acoustics, Speech, and Signal Processing, (ICASSP)*, 1999.
- [11] A. Doucet and X. Wang, “Monte Carlo methods for signal processing,” *IEEE Signal Processing Magazine*, vol. 22, no. 6, pp. 152–170, Nov. 2005.
- [12] 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.
- [13] L. Holden, R. Hauge, and M. Holden, “Adaptive independent Metropolis-Hastings,” *The Annals of Applied Probability*, vol. 19, no. 1, pp. 395–413, 2009.
- [14] L. Martino, R. Casarin, F. Leisen, and D. Luengo, “Adaptive sticky generalized Metropolis,” Tech. Rep., Sep. 2013, arXiv:1308.3779v2.
- [15] B. Cai, R. Meyer, and F. Perron, “Metropolis-Hastings algorithms with adaptive proposals,” *Statistics and Computing*, vol. 18, pp. 421–433, 2008.
- [16] Y. Fan, S. Brooks, and A. Gelman, “Output assessment for Monte Carlo simulations via the score statistic,” *Journal of Computational and Graphical Statistics*, vol. 15, pp. 178–206, 2006.