Comput Stat
DOI 10.1007/s00180-016-0643-9

ORIGINAL PAPER

Issues in the Multiple Try Metropolis mixing

L. Martino[1] · **F. Louzada**[1]

Received: 29 August 2015 / Accepted: 20 January 2016

Abstract The Multiple Try Metropolis (MTM) algorithm is an advanced MCMC technique based on drawing and testing several candidates at each iteration of the algorithm. One of them is selected according to certain weights and then it is tested according to a suitable acceptance probability. Clearly, since the computational cost increases as the employed number of tries grows, one expects that the performance of an MTM scheme improves as the number of tries increases, as well. However, there are scenarios where the increase of number of tries does not produce a corresponding enhancement of the performance. In this work, we describe these scenarios and then we introduce possible solutions for solving these issues.

Keywords Multiple Try Metropolis algorithm · Multi-point Metropolis algorithm · MCMC methods · MTM with variable number of tries

1 Introduction

Markov chain Monte Carlo (MCMC) methods are classical Monte Carlo techniques (Robert and Casella 2004), that produce a Markov chain converging to a target probability density function (pdf), usually to approximate an otherwise-incalculable integral (Liu 2004; Liang et al. 2010).

The Multiple-Try Metropolis (MTM) method (Liu et al. 2000) is an extension of the Metropolis-Hastings algorithm (Metropolis et al. 1953; Hastings 1970) in which the next state of the chain is selected among a set of N independent and identically

✉ L. Martino
lukatotal@gmail.com

[1] Institute of Mathematical Sciences and Computing, Universidade de São Paulo, São Carlos, São Paulo, Brazil

Published online: 12 February 2016

distributed (i.i.d.) samples. This enables the MTM sampler to make large step-size jumps without a lowering in the acceptance rate; and thus MTM can explore easily a larger portion of the sample space in fewer iterations. Different MTM schemes have been proposed in literature (Frenkel and Smit 1996, Chapter 13), (Qin and Liu 2001; Casarin et al. 2013; Pandolfi et al. 2010; Martino et al. 2012; Craiu and Lemieux 2007) and have been studied in several works (Bédard et al. 2012; Martino and Read 2013; Martino et al. 2014). More recently parallel MTM algorithms have been proposed in Martino et al. (2015a).

A well-designed MTM scheme improves its performance as the number of tries, N, grows. Namely, when N grows approaching infinity, the correlation among the generated samples should vanish to zero. Clearly, this is at the expense of an increasing computational cost due to the use of a greater number of tries. In this work, we describe certain scenarios where the use of a greater N in a standard MTM method (Liu et al. 2000) and its extensions (Casarin et al. 2013; Pandolfi et al. 2010; Martino et al. 2012; Martino and Read 2013) does not yield an improvement in the performance. We explain the reasons of these drawbacks, and provide possible solutions for fixing these issues. The first scenario involves the use of a single random-walk proposal within a standard MTM structure, whereas, in the second scenario, the use of multiple proposal pdfs independent from the previous state of the chains is considered. In the first one, the increase of number of tries is always prejudicial, regardless of the choice of the weight functions [involving the target function in a suitable way (Liu et al. 2000; Martino and Read 2013)]. In the second one, the increase of number of tries can help the mixing of the chain using a certain class of the weight functions (clearly, at the expense of a greater computational cost). However, we discuss different ways of using the set of multiple independent proposal pdfs within an MTM scheme improving the performance, in any case. For improving the performance in the first scenario, we suggest to use an MTM with variable number of tries, in a suitable way without jeopardizing the ergodicity of the chain.

2 Multiple Try Metropolis with a single random-walk proposal

Let us denote the target density as $\bar{\pi}(\mathbf{x}) \propto \pi(\mathbf{x})$. First of all, we consider the use of a single random-walk proposal density, $q(\mathbf{z}|\mathbf{x}_{t-1}) = q(\mathbf{z} - \mathbf{x}_{t-1})$. Given a current state of the chain $\mathbf{x}_{t-1} \in \mathcal{X} \subseteq \mathbb{R}^{d_X}$, $t \in \mathbb{N}$, an MTM scheme generates N independent candidates $\{\mathbf{z}_1, \ldots, \mathbf{z}_N\}$ from a proposal density q, i.e.,

$$\mathbf{z}_1, \ldots, \mathbf{z}_N \sim q(\mathbf{z}|\mathbf{x}_{t-1}).$$

Then, one sample $\mathbf{z}$ is selected among the set $\{\mathbf{z}_1, \ldots, \mathbf{z}_N\}$, according to certain weight functions (Liu et al. 2000; Martino and Read 2013). The movement from $\mathbf{x}_t$ to $\mathbf{z}$ is accepted with a suitable probability $\alpha(\mathbf{x}_{t-1}, \mathbf{z})$, which also depends on the rest of candidates. The probability $\alpha(\mathbf{x}_{t-1}, \mathbf{z})$ is designed such that the kernel of the MTM algorithm fulfills the detailed balance condition. Only for facilitating the comprehension, we consider the importance weights

Table 1 Multiple Try Metropolis with a (single) random-walk proposal (RW-MTM)

1. Draw N independent samples from the proposal pdf, $\mathbf{z}_1, \ldots, \mathbf{z}_N \sim q(\mathbf{x}|\mathbf{x}_{t-1}) = q(\mathbf{z} - \mathbf{x}_{t-1})$
2. Select a sample $\mathbf{z} \in \{\mathbf{z}_1, \ldots, \mathbf{z}_N\}$, according to the probabilities
$$\bar{w}_k = \frac{w(\mathbf{z}_k|\mathbf{x}_{t-1})}{\sum_{n=1}^{N} w(\mathbf{z}_n|\mathbf{x}_{t-1})}, \quad \text{where} \quad w(\mathbf{z}_k|\mathbf{x}_{t-1}) = \frac{\pi(\mathbf{z}_k)}{q(\mathbf{z}_k|\mathbf{x}_{t-1})}, \quad (1)$$
for $k = 1, \ldots, N$
3. Draw $N-1$ auxiliary points from the proposal q given the previous selected sample $\mathbf{z}$, namely $\mathbf{y}_1, \ldots, \mathbf{y}_{N-1} \sim q(\mathbf{x}|\mathbf{z})$, and set $\mathbf{y}_N = \mathbf{x}_{t-1}$
4. Compute the weights of the auxiliary points,
$$w(\mathbf{y}_k|\mathbf{z}) = \frac{\pi(\mathbf{y}_k)}{q(\mathbf{y}_k|\mathbf{z})}, \quad \text{for} \quad k = 1, \ldots, N. \quad (2)$$
5. Set $\mathbf{x}_t = \mathbf{z}$ with probability
$$\alpha(\mathbf{x}_{t-1}, \mathbf{z}) = \min\left[1, \frac{\sum_{n=1}^{N} w(\mathbf{z}_n|\mathbf{x}_{t-1})}{\sum_{n=1}^{N} w(\mathbf{y}_n|\mathbf{z})}\right] \quad (3)$$
Otherwise, set $\mathbf{x}_t = \mathbf{x}_{t-1}$, with probability $1 - \alpha(\mathbf{x}_{t-1}, \mathbf{z})$

$$w(\mathbf{z}_k|\mathbf{x}_{t-1}) = \frac{\pi(\mathbf{z}_k)}{q(\mathbf{z}_k|\mathbf{x}_{t-1})}, \quad (4)$$

for choosing $\mathbf{z} \in \{\mathbf{z}_1, \ldots, \mathbf{z}_N\}$, i.e., $\mathbf{z}$ is selected according the probabilities $\bar{w}_k = \frac{w(\mathbf{z}_k|\mathbf{x}_{t-1})}{\sum_{n=1}^{N} w(\mathbf{z}_n|\mathbf{x}_{t-1})}$. Different kind of weights could be used (Martino and Read 2013; Pandolfi et al. 2010), but without avoiding the problem that we describe in the next section.

Table 1 shows all the details of the MTM technique. Observe that, an RW-MTM method requires the generation of $N-1$ auxiliary points $\mathbf{y}_1, \ldots, \mathbf{y}_{N-1}$ from $q(\cdot|\mathbf{z})$ (see Step 3 of Table 1). Moreover, note that the selected sample $\mathbf{z}$ is drawn from the empirical measure

$$\hat{\pi}^{(N)}(\mathbf{z}) = \sum_{n=1}^{N} \bar{w}_n \delta(\mathbf{z} - \mathbf{z}_n), \quad (5)$$

that approximates the distribution of π, via importance sampling (IS) (Robert and Casella 2004; Liu 2004). Finally, we remark that the acceptance probability $\alpha(\mathbf{x}_{t-1}, \mathbf{z})$ in Eq. (3) can be expressed as

$$\alpha(\mathbf{x}_{t-1}, \mathbf{z}) = \min\left[1, \frac{\hat{Z}(\mathbf{z}_1, \ldots, \mathbf{z}_N|\mathbf{x}_{t-1})}{\hat{Z}(\mathbf{y}_1, \ldots, \mathbf{y}_N|\mathbf{z})}\right], \quad (6)$$

where the function $\hat{Z}(\cdot|\mathbf{r}) : \mathcal{X}^N \to \mathbb{R}$, with $\mathbf{r} \in \mathcal{X}$,

$$\hat{Z}(\mathbf{v}_1, \ldots, \mathbf{v}_N|\mathbf{r}) = \frac{1}{N} \sum_{n=1}^{N} \frac{\pi(\mathbf{v}_n)}{q(\mathbf{v}_n|\mathbf{r})}, \quad (7)$$

is an estimator of the normalizing constant $Z = \int_{\mathcal{X}} \pi(\mathbf{x})d\mathbf{x}$ (Robert and Casella 2004), i.e., of the area below $\pi(\mathbf{x})$.

3 Problem in the RW-MTM mixing

The desired behavior of an MTM scheme is that the performance improves as the number of used candidates N grows (jointly with the computational cost). Indeed in general, as N increases, the chosen point $\mathbf{z}$ is selected from a better IS approximation $\hat{\pi}^{(N)}$ of $\bar{\pi}$, so that $\mathbf{z}$ is a better candidate to be tested as new possible state of the chain. As a consequence, in a well-designed MTM scheme the acceptance probability $\alpha(\mathbf{x}_{t-1}, \mathbf{z})$ should approach 1 when $N \to \infty$. Thus, in general, MTM fosters greater "jumps" and, as a consequence, a faster exploration of the state space. However, below we describe a scenario where the increase of number N of tries could be even damaging.

For facilitating the explanation, we assume that the expected value of the random variable $\mathbf{Z} \sim q(\mathbf{z} - \mathbf{x}_{t-1})$ is exactly $\mathbf{x}_{t-1}$, i.e., $E[\mathbf{Z}] = \mathbf{x}_{t-1}$, e.g., when q is Gaussian, $q(\mathbf{z} - \mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{z}; \mathbf{x}_{t-1}, \mathbf{C})$. Let us denote $\hat{Z}_1 = \hat{Z}(\mathbf{z}_1, \ldots, \mathbf{z}_N|\mathbf{x}_{t-1})$ and $\hat{Z}_2 = \hat{Z}(\mathbf{y}_1, \ldots, \mathbf{y}_N|\mathbf{z})$, so that we can rewrite the acceptance probability as

$$\alpha = \min\left[1, \frac{\hat{Z}_1}{\hat{Z}_2}\right]. \tag{8}$$

Furthermore, consider a scenario where the state in the $(t-1)$-th iteration, $\mathbf{x}_{t-1}$, is placed in a region of low probability of $\bar{\pi}(\mathbf{x}) \propto \pi(\mathbf{x})$, nearby a region of high probability mass (e.g., see Fig. 1a). Assume also that the variance of the proposal

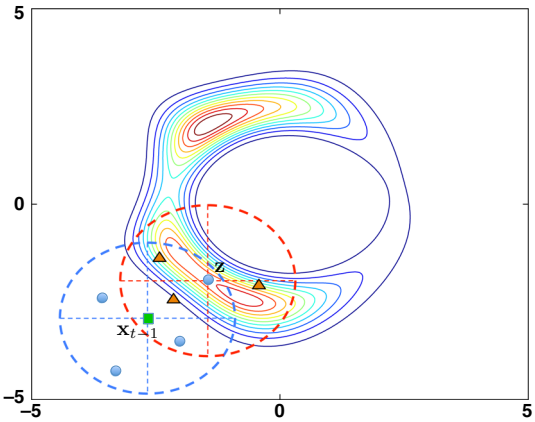

Fig. 1 Graphical representation of a possible scenario described in Sect. 3, where $\hat{Z}_2 > \hat{Z}_1$ (and $\hat{Z}_2 >> \hat{Z}_1$ when N grows). We show the contour plot of a bidimensional target pdf $\pi(\mathbf{x})$ with *solid lines*. The previous state of the chain $\mathbf{x}_{t-1}$ is depicted with a *square*; the $N = 4$ candidates $\mathbf{z}_j$'s are shown with *circles*, whereas the $N-1 = 3$ auxiliary points $\mathbf{y}_i$'s are illustrated with *triangles*. *Dashed lines* represent the scale parameters of the proposal densities $q(\cdot|\mathbf{x}_{t-1})$ and $q(\cdot|\mathbf{z})$, where $\mathbf{z} \in \{\mathbf{z}_1, \ldots, \mathbf{z}_4\}$ is the selected candidate

$q(\mathbf{z}-\mathbf{x}_{t-1})$ is wide enough in order to (at least) reach the region of high probability mass of π. In this situation, several drawn tries are located in the region of small probability around the value $E[\mathbf{Z}]=\mathbf{x}_{t-1}$. On the other hand, it is possible that few of them are located close to the mode of π; Fig. 1a depicts a possible scenario of this kind, with only $N=4$ tries and one of them located in a mode of π. Thus, it is highly probable that the MTM selected one well-located point as proposed sample $\mathbf{z}$, after the resampling at Step 2. For the same reasons, in general, many of the $N-1$ auxiliary points, $\mathbf{y}_1,\ldots,\mathbf{y}_{N-1}$ drawn from $q(\mathbf{y}|\mathbf{z})$, will be placed around the mode of π. Hence, in this situation, we have that

$$\hat{Z}_2=\frac{1}{N}\sum_{n=1}^{N}\frac{\pi(\mathbf{y}_n)}{q(\mathbf{y}_n|\mathbf{z})} >> \hat{Z}_1=\frac{1}{N}\sum_{n=1}^{N}\frac{\pi(\mathbf{z}_n)}{q(\mathbf{z}_n|\mathbf{x}_{t-1})}.$$

As a consequence,

$$\alpha(\mathbf{x}_{t-1},\mathbf{z})\approx 0,$$

so that the chain can remain stuck at $\mathbf{x}_{t-1}$. It is important to observe that this situation can become even worse if N grows. On the contrary, in this scenario, the use of a smaller number of tries can help to jump to the region of high probability. Finally, we remark that the problem previously described cannot be solved by changing of analytical form of the weights (Liu et al. 2000; Martino and Read 2013).[1]

3.1 Proposed solution

Let us denote as $K_m(\mathbf{x}_t|\mathbf{x}_{t-1},N_m)$ the kernel of an MTM scheme employing N_m tries. We consider a combination M different kernels each of which using a different number of tries N_m, $m=1,\ldots,M$, i.e.,

$$K(\mathbf{x}_t|\mathbf{x}_{t-1})=\frac{1}{M}\sum_{m=1}^{M}K_m(\mathbf{x}_t|\mathbf{x}_{t-1},N_m). \tag{9}$$

It is straightforward to show that if each $K_m(\mathbf{x}_t|\mathbf{x}_{t-1},N_m)$ leaves invariant π, also $K(\mathbf{x}_t|\mathbf{x}_{t-1})$ has π as invariant pdf (Robert and Casella 2004; Liu 2004). Therefore, fixing the averaged computational effort, represented by the averaged number or tries

$$\widetilde{N}=\frac{1}{M}\sum_{m=1}^{M}N_m,$$

we choose M different values $N_m\in\mathbb{N}$, such that $\widetilde{N}$ is the desired one. The idea is to use a variable number of tries, i.e., a different number of candidates at each iteration.

[1] A suitable acceptance function α for generic weight functions is shown in "Appendix", for the case of multiple independent proposal densities.

Namely, at each iteration, an index m' is drawn uniformly within $1, \ldots, M$ and then $N_{m'}$ tries are employed in the MTM scheme $K_{m'}$. Note that this is equivalent to use the kernel in Eq. (9). Choosing at least one small value, e.g., $N_1 = 1$, this helps jumps of the chain in the awkward scenario, previously described. See the numerical simulations for further details.

4 Multiple Try Metropolis with different independent proposals

The MTM algorithm in Table 1 can be simplified if the proposal pdf $q(\mathbf{x})$ is independent from the previous state of the generated chain. Indeed, in this case, Step 3 in Table 1 can be removed, in the sense that it is possible to avoid the generation of the auxiliary points (Liu et al. 2000; Martino and Read 2013). Furthermore, it is also possible to employ simultaneously different proposal pdfs $q_1(\mathbf{x}), \ldots, q_N(\mathbf{x})$ (Casarin et al. 2013; Martino and Read 2013). The resulting algorithm is detailed in Table 2, considering the use of importance weights. The acceptance probability α in Eq. (12) can be written again as

$$\alpha = \min\left[1, \frac{\hat{Z}_1}{\hat{Z}_2}\right],$$

where, in this case,

$$\begin{aligned} \hat{Z}_1 &= \frac{1}{N}\sum_{n=1}^{N} w_n(\mathbf{z}_n), \\ \hat{Z}_2 &= \frac{1}{N}\left(N\hat{Z}_1 - w_j(\mathbf{z}_j) + w_j(\mathbf{x}_{t-1})\right). \end{aligned} \tag{10}$$

The general acceptance function α for I-MTM using generic (bounded and positive) weights is shown in Eq. (14).

5 Problem in the I-MTM mixing

First of all, we can observe that the sums in $\hat{Z}_1$ and $\hat{Z}_2$ in Eq. (10) differ only for one weight, i.e., $\hat{Z}_1$ contains $w_j(\mathbf{z}_j)$ but does not involve $w_j(\mathbf{x}_{t-1})$, whereas $\hat{Z}_2$ includes $w_j(\mathbf{x}_{t-1})$, instead of $w_j(\mathbf{z}_j)$. Thus, using importance weights, the probability α of an I-MTM scheme always approaches 1 when N increases, if the employed weight functions are included in the class of weights proposed in Liu et al. (2000).[2] This statement is instead not valid, in general, for the generic weight functions given in Pandolfi et al. (2010), Martino and Read (2013) and recalled in Eq. (14).

In this section we focus on the use of importance weights, which are contained in class discussed in Liu et al. (2000). The solutions that we discuss later on are valid

[2] Considering the case of independent proposal pdfs, the class of weights in Liu et al. (2000) is defined as $w_k(\mathbf{y}_k|\mathbf{z}) = \pi(\mathbf{z}_k)q_k(\mathbf{x})\lambda_k(\mathbf{z}_k, \mathbf{x})$ with $k = 1, \ldots, N$, and $\lambda_k(\mathbf{z}_k, \mathbf{x}) = \lambda_k(\mathbf{x}, \mathbf{z}_k)$ is a generic symmetric function w.r.t. $\mathbf{z}_k$ and $\mathbf{x}$. As an example, if we set $\lambda_k(\mathbf{z}_k, \mathbf{x}) = \frac{1}{q_k(\mathbf{x})q_k(\mathbf{z}_k)}$, we obtain the importance weights $w_k(\mathbf{z}_k|\mathbf{x}) = w_k(\mathbf{z}_k) = \frac{\pi(\mathbf{z}_k)}{q_k(\mathbf{z}_k)}$.

Table 2 Multiple Try Metropolis with different independent proposals (I-MTM)

1. Draw N independent samples
   $\mathbf{z}_1 \sim q_1(\mathbf{x}), \ldots, \mathbf{z}_N \sim q_N(\mathbf{x})$
2. Select a sample $\mathbf{z}_j \in \{\mathbf{z}_1, \ldots, \mathbf{z}_N\}$, according to the probabilities
   $$\bar{w}_k = \frac{w_k(\mathbf{z}_k)}{\sum_{n=1}^{N} w_n(\mathbf{z}_n)}, \quad \text{where} \quad w_k(\mathbf{z}_k) = \frac{\pi(\mathbf{z}_k)}{q_k(\mathbf{z}_k)}, \qquad (11)$$
   for $k = 1, \ldots, N$
3. Set $\mathbf{x}_t = \mathbf{z}_j$ with probability
   $$\alpha(\mathbf{x}_{t-1}, \mathbf{z}_j) = \min\left[1, \frac{\sum_{n=1}^{N} w_n(\mathbf{z}_n)}{\sum_{n=1}^{N} w_n(\mathbf{z}_n) - w_j(\mathbf{z}_j) + w_j(\mathbf{x}_{t-1})}\right] \qquad (12)$$
   Otherwise, set $\mathbf{x}_t = \mathbf{x}_{t-1}$, with probability $1 - \alpha(\mathbf{x}_{t-1}, \mathbf{z}_j)$

in any cases, including the use of the generic weights in "Appendix". Note that, in I-MTM, the j-th weight involves the j-th proposal pdf, i.e.,

$$w_j(\mathbf{x}) = \frac{\pi(\mathbf{x})}{q_j(\mathbf{x})}.$$

We need to evaluate the j-th weight w_j, involving the j-th proposal q_j, at $\mathbf{z}_j$ and $\mathbf{x}_{t-1}$. The sample $\mathbf{z}_j$ is drawn from q_j by definition, whereas $\mathbf{x}_{t-1}$ is the previous state of the chain (it could be generated from any possible q_n in the previous iterations of the I-MTM algorithm). Hence, with high probability $\mathbf{z}_j$ is located nearby a mode of q_j, since $\mathbf{z}_j \sim q_j(\mathbf{z})$, whereas $\mathbf{x}_{t-1}$ could be placed close to a mode or a tail of q_j with equal chance, in general. Thus, since the proposal q_j appears in the denominator of the weights w_j, in general we have $w_j(\mathbf{z}_j) < w_j(\mathbf{x}_{t-1})$, producing small values of acceptance probability α, if N is not enough big. This scenario becomes even more complicated, if the proposal pdf q_j is placed close to a mode of the target π, and the previous state $\mathbf{x}_{t-1}$ is located in a tail of q_j. In this case, if $\pi(\mathbf{x}_{t-1}) \neq 0$, the value of $w_j(\mathbf{x}_{t-1})$ can be huge and $w_j(\mathbf{x}_{t-1}) >> w_j(\mathbf{z}_j)$. Hence, the I-MTM scheme tends to select several times the sample drawn from q_j, i.e., $\mathbf{z}_j$, as "good" candidate (step 2 of Table 2), but the movement from $\mathbf{x}_{t-1}$ to $\mathbf{z}_j$ is often rejected since $\alpha \approx 0$. As a consequence, the chain can remain indefinitely trapped in this situation. Figure 2 represents graphical sketch of this situation.

5.1 Proposed solutions

Below, we discuss different possible solutions, ordered for increasing theoretical complexity and practical interest. It is important to remark that the change of the analytic form of the weights is not a solution as shown in "Appendix".

5.1.1 First solution

First of all, let us consider the possibility of using a greater number of tries keeping fixed the number N of proposal pdfs, i.e., denoting with P the number of tries we

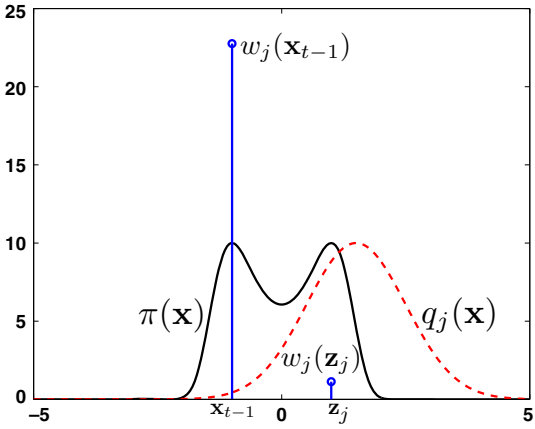

Fig. 2 Graphical representation of the scenario described in Sect. 5. The contour plot of a bimodal (unnormalized) target pdf $\pi(\mathbf{x})$ is depicted with *solid line* whereas the j-th (unnormalized) proposal pdf $q_j(\mathbf{x})$ is shown with *dashed line*

have $P > N$ with $P = kN$ with $k \in \mathbb{N}$. The problem described above could be solved increasing P, when the used weights are importance weights.[3] If $\mathbf{x}_{t-1}$ is located in a tail of q_j, the value of P required to solve the issue, could be huge. However, this trivial solution entails an increase of the computational cost in terms of evaluations of the target function. In the sequel, we introduce alternative solutions which do not require to increase the computational cost and are valid for any possible kind of weight functions, used within I-MTM.

5.1.2 Second solution

The problem described above disappears if we consider a unique proposal pdf defined as mixture, i.e.,

$$\psi(\mathbf{x}) = \frac{1}{N}\sum_{n=1}^{N} q_n(\mathbf{x}).$$

Hence, in this case, we draw $\mathbf{z}_1, \ldots, \mathbf{z}_N$ from $\psi(\mathbf{x})$ and the weights are

$$w(\mathbf{z}_n) = \frac{\pi(\mathbf{z}_n)}{\psi(\mathbf{z}_n)}.$$

We can observe that in the denominator of the importance weight all the components q_n's are used and hence evaluated, in this case. Let us assume that the previous state of the chain $\mathbf{x}_{t-1}$ was generated from the k-th component of the mixture, i.e., $q_k(\mathbf{x})$, in a previous iteration, and the selected candidate z_j has been drawn from $q_j(\mathbf{x})$, by definition. In this scenario, both pdfs, q_k and q_j, are involved simultaneously in the denominator of importance weights, avoiding the problem previously described. Although the mixture $\psi(\mathbf{x})$ takes into account all the proposal pdfs q_n's, unlike in the

[3] When other kind of weights is employed, the problem could persist even increasing P.

I-MTM in Table 2, in this case only a subset of the components $\{q_1(\mathbf{x}), \ldots, q_N(\mathbf{x})\}$ participates in generating candidates at each iteration. To avoid this drawback, see below the next solution.

5.1.3 Third solution

The joint use of the functions $q_1(\mathbf{x}), \ldots, q_N(\mathbf{x})$ (with equal proportion, at each iteration) in general increases the robustness of the resulting algorithm. Namely, if no information is available to choose the best proposal in the set $\{q_1(\mathbf{x}), \ldots, q_N(\mathbf{x})\}$, a more robust strategy consists in employing always the complete set of functions. The *deterministic mixture* (DM) approach (Veach and Guibas 1995; Owen and Zhou 2000; Elvira et al. 2015a,b), successfully applied in different sophisticated Monte Carlo algorithms (Cornuet et al. 2012; Martino et al. 2015b,c), provides a possible solution. Indeed, using the DM approach, we can draw one sample $\mathbf{z}_n$ from each proposal pdf $q_n(\mathbf{x})$, i.e.,

$$\mathbf{z}_1 \sim q_1(\mathbf{x}), \ldots, \mathbf{z}_N \sim q_N(\mathbf{x}),$$

exactly as in step 1 of Table 2, and then assign the corresponding DM weights

$$w(\mathbf{z}_n) = \frac{\pi(\mathbf{z}_n)}{\psi(\mathbf{z}_n)} = \frac{\pi(\mathbf{z}_n)}{\frac{1}{N}\sum_{n=1}^{N} q_n(\mathbf{x})}, \quad n = 1, \ldots, N.$$

It is possible to show that this approach is valid and it can be interpreted as variance reduction technique for sampling from a mixture of pdfs. Namely, we use a quasi-Monte Carlo approach for generating the indices j_n, $n = 1, \ldots, N$, i.e., the deterministic sequence $j_1 = 1$, $j_2 = 2, \ldots, j_n = N$, and then $\mathbf{z}_n \sim p(\mathbf{x}|j_n) = q_n(\mathbf{x})$ for $n = 1, \ldots, N$. The DM approach improves the performance of the IS numerical approximation (Owen and Zhou 2000; Elvira et al. 2015a). Observe that, also in this case, we solve the issue, since again all the proposals are included in the denominator of the weights, and we always use all the proposals $q_1, \ldots, q_N$ at each iteration (as in Table 2).

6 Numerical simulations: localization in a wireless sensor network

We consider the problem of positioning a target $\mathbf{X}$ in a two-dimensional space using range measurements (Ali et al. 2007; Fitzgerald 2001). More formally, we consider a random vector $\mathbf{X} = [X_1, X_2]^\top$ denoting the target's position in $\mathbb{R}^2$. The measurements are obtained from 6 sensors located at $\mathbf{h}_1 = [-5, 1]^\top$, $\mathbf{h}_2 = [-2, 6]^\top$, $\mathbf{h}_3 = [0, 0]^\top$, $\mathbf{h}_4 = [5, -6]^\top$, $\mathbf{h}_5 = [6, 4]^\top$ and $\mathbf{h}_6 = [-4, -4]^\top$, and the observation equations are given by

$$R_j = -10\log\left(\frac{||\mathbf{X} - \mathbf{h}_j||}{0.3}\right) + \Omega_j, \quad j = 1, \ldots, 6, \tag{13}$$

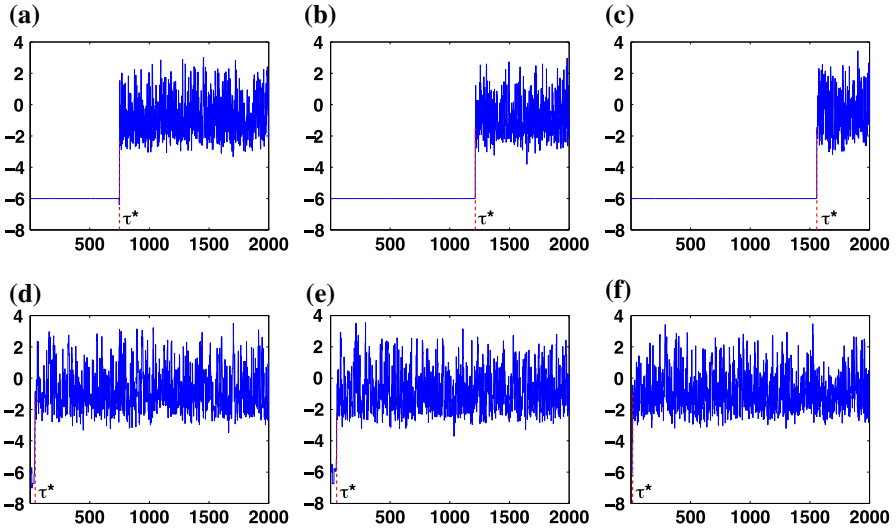

Fig. 3 **a**–**c** Realizations of the standard RW-MTM method with **a** $N = \widetilde{N} = 200$ ($\tau^* = 750$, in this specific run), **b** $N = \widetilde{N} = 500$ ($\tau^* = 1214$) and **c** $N = \widetilde{N} = 1000$ ($\tau^* = 1558$). **d**—**f** Realizations of the novel method with **d** $\widetilde{N} = 200$ ($\tau^* = 43$, in this run), **e** $\widetilde{N} = 500$ ($\tau^* = 52$) and **f** $\widetilde{N} = 1000$ ($\tau^* = 15$)

where Ω_j are i.i.d. Gaussian random variables, $\Omega_j \sim \mathcal{N}(\omega_j; 0, 5)$. Let us assume to receive the observation vector $\mathbf{r} = [26, 26.5, 25, 28, 28, 25.3]^\top$. In order to perform Bayesian inference, we consider a non-informative prior over $\mathbf{X}$ (i.e., an improper uniform density on $\mathbb{R}^2$), and study the posterior pdf, $\bar{\pi}(\mathbf{x}) = p(\mathbf{x}|\mathbf{r}) \propto p(\mathbf{r}|\mathbf{x})p(\mathbf{x})$. A contour plot of $\bar{\pi}(\mathbf{x}) \propto \pi(\mathbf{x})$ is shown in Fig. 1.

We perform different MTM schemes for drawing samples from the posterior $\bar{\pi}(\mathbf{x})$. In order to highlight the described issues, we decide the starting point of the chain at $\mathbf{x}_0 = [-6, -6]^\top$ forcing the chain to escape from a region of low probability of $\bar{\pi}(\mathbf{x})$. We run 500 independent simulations of different MTM schemes with $t = 1, \ldots, T$ (we set $T = 2000$ for RW-MTM and $T = 4000$ for I-MTM), and compute the expected time needed for the chain to escape from the region around $\mathbf{x}_0$ and reach the region containing the modes of the target. For this purpose, at each iteration of the algorithm, we calculate the Euclidean distances $d_{1,t} = ||\mathbf{x}_t - \mathbf{x}_0||$ and $d_{2,t} = ||\mathbf{x}_t - \boldsymbol{\mu}||$ where $\boldsymbol{\mu} = E_\pi[\mathbf{X}] = [-0.753, -0.037]^\top$ is the expected value of $\mathbf{X} \sim \bar{\pi}(\mathbf{x})$.[4] At each run, we obtain the first iteration τ^* such that $d_{1,\tau^*} > d_{2,\tau^*}$, hence τ^* can be interpreted as the time that the chain remained trapped around $\mathbf{x}_0$, in the specific run (see Fig. 3 as examples of τ^*). Cleary, we have $1 \leq \tau^* \leq T$. We repeat the procedure for 500 independent runs, in order to approximate the expected time $E[\tau^*]$.

6.1 RW-MTM

For the random walk MTM method, we consider a Gaussian proposal $q(\mathbf{x}|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}; \mathbf{x}_{t-1}, \boldsymbol{\Sigma})$ where $\boldsymbol{\Sigma} = \sigma^2 \mathbb{I}_2$ with $\sigma \in \{0.5, 0.8, 1\}$. We test different averaged

[4] We have computed the vector $E_\pi[\mathbf{X}]$ numerically, using a computational expensive thin grid in $\mathbb{R}^2$.

Table 3 Expected number of iterations $E[\tau^*]$ required to escape from the region around $\mathbf{x}_0 = [-6, -6]^\top$ with RW-MTM

Scheme	σ	$\widetilde{N} = 50$	$\widetilde{N} = 100$	$\widetilde{N} = 200$	$\widetilde{N} = 500$	$\widetilde{N} = 1000$
Standard	0.5	101.922	165.320	276.454	431.606	601.050
Novel		67.237	72.349	81.253	92.798	88.444
Standard	0.8	205.299	367.358	612.442	1098.5	1363.1
Novel		49.711	51.557	49.405	49.706	56.145
Standard	1	237.326	443.080	709.808	784.644	699.614
Novel		43.436	41.236	33.906	37.812	39.270

Table 4 MSE in the estimation of $E_\pi[\mathbf{X}]$, obtained by RW-MTM, with $\sigma = 1$ and $\mathbf{x}_0 \sim \mathcal{U}([-6, 6] \times [-6, 6])$, i.e., randomly chosen at each run

Scheme	$\widetilde{N} = 50$	$\widetilde{N} = 100$	$\widetilde{N} = 200$	$\widetilde{N} = 500$	$\widetilde{N} = 1000$
Standard	0.1702	0.1193	0.0892	0.0542	0.0266
novel	0.0533	0.0428	0.0329	0.0320	0.0228

The standard and the novel scheme are test with different (fixed or averaged) number of tries $\widetilde{N}$

number of tries $\widetilde{N} \in \{50, 100, 200, 500, 1000\}$. Thus, in the standard RW-MTM scheme, we set $N = \widetilde{N}$, whereas in the proposed mixture of MTM kernels in Eq. (9), we consider $M = 3$ and $N_1 = 1$, $N_2 = \widetilde{N}$, $N_3 = 2\widetilde{N} - 1$, so that we have always

$$\widetilde{N} = \frac{N_1 + N_2 + N_3}{3}.$$

Therefore, the averaged computational cost is the same in both schemes, in terms of evaluations of the target distribution. The results, in terms of the expected number of iterations $E[\tau^*]$, are provided in Table 3. First of all, observe that, in general, $E[\tau^*]$ grows if the number of tries N increases especially for the standard RW-MTM method (recall that for the standard RW- MTM scheme $N = \widetilde{N}$). The expected number of iterations $E[\tau^*]$ of the novel MTM technique with variable number of tries (introduced in Sect. 3.1) is always smaller than the corresponding value of the standard RW-MTM method. Namely, the novel scheme always outperforms the standard one, escaping from the region around $\mathbf{x}_0$ and reaching the modes of $\bar{\pi}(\mathbf{x})$ more quickly, whereas the standard RW-MTM method remains stuck around $\mathbf{x}_0$ for several iterations, prejudicing its performance. Figure 3 shows the improvement in the mixing with the proposed solution with respect to the standard RW-MTM technique.

Furthermore, the Mean Square Error (MSE) in the estimation of $E_\pi[\mathbf{X}]$ obtained by RW-MTM (and averaged over 500 runs) is provided in Table 4. In this case, we set $\sigma = 1$ and the initial state is chosen randomly $\mathbf{x}_0 \sim \mathcal{U}([-6, 6] \times [-6, 6])$ (i.e., uniformly in the square $([-6, 6] \times [-6, 6])$, at each run. We can observe that the novel scheme provides always the smallest MSE confirming the robustness of the proposed solution.

Table 5 Expected number of iterations $E[\tau^*]$ required to escape from the region around $\mathbf{x}_0 = [-6, -6]^\top$ with I-MTM

Scheme	Conf	$\sigma = 1.25$	$\sigma = 1.3$	$\sigma = 1.35$	$\sigma = 1.4$
Standard	1	2967.6	1185.6	128.102	15.610
Novel		7.338	10.198	13.652	10.834
Standard	2	3015.6	1212.9	139.816	20.548
Novel		10.130	20.454	6.989	15.920

Table 6 MSE in the estimation of $E_\pi[\mathbf{X}]$, obtained by I-MTM, with **Conf2** and $\mathbf{x}_0 \sim \mathcal{U}([-6, 6] \times [-6, 6])$, i.e., randomly chosen at each run

Scheme	$\sigma = 1.25$	$\sigma = 1.3$	$\sigma = 1.35$	$\sigma = 1.4$
Standard	6.7943	6.4345	5.9183	5.5595
Novel	0.7677	0.6987	0.3135	0.3055

6.2 I-MTM

For the I-MTM scheme, we consider $N = 2$ proposal pdfs and also $P = N = 2$ number of tries (exactly as in the algorithm described in Table 2). Furthermore, the proposal pdfs are both Gaussians, specifically, $q_n(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_n, \boldsymbol{\Sigma})$, for $n = 1, 2$ and $\boldsymbol{\mu}_1 = [-6, -6]^\top$, $\boldsymbol{\mu}_2 = [0, 0]^\top$ in the first configuration (denoted as **Conf1**), and $\boldsymbol{\mu}_1 = [-6, -6]^\top$, $\boldsymbol{\mu}_2 = [-1, -2]^\top$ in a second one (denoted as **Conf2**). Thus, the second proposal pdf is always well-located, unlike the first one. The covariance matrix is the same for both proposals, $\boldsymbol{\Sigma} = \sigma^2 \mathbb{I}_2$, and we test several values of σ,, i.e., $\sigma \in \{1.25, 1.3, 1.35, 1.4\}$. As alternative scheme we consider the use of the deterministic mixture approach proposed in Sect. 5.1. We compute again the expected number of iterations $E[\tau^*]$ for reaching the modes starting from $\mathbf{x}_0 = [-6, -6]^\top$ and set $T = 4000$ as length of the chain, in this case. The results are provided in Table 5. We can observe that with the deterministic mixture approach the chain is able to jump easily to the regions of high probability of π, unlike with the standard I-MTM scheme. This occurs for every value of σ. With the standard I-MTM scheme the chain remains trapped around $\mathbf{x}_0$ for several iterations jeopardizing the performance of the algorithm (see also Table 6).

The MSE values given in Table 6 (and averaged over 500 runs) show that the improvement obtained by the novel scheme is even more evident than in the RW-MTM case. We have considered **Conf2** and the initial state is chosen randomly $\mathbf{x}_0 \sim \mathcal{U}([-6, 6] \times [-6, 6])$ at each run.

7 Conclusions

In this work, we have described different scenarios where MTM schemes have not the desired behavior, preventing the fast exploration of the state space. These drawbacks cannot be solved simply increasing the computational effort, in terms of used number of tries. We have restricted the description of the problematic cases considering only the

importance weights for the sake of simplicity, but the issues persist with other generic weight functions. Furthermore, we provide and discuss different solutions that solved the previously described problems, as also shown with numerical simulations. The proposed MTM schemes are in general more robust than the corresponding standard MTM techniques.

Acknowledgments We would like to thank the Reviewers for their comments which have helped us to improve the manuscript. This work has been supported by the Grant 2014/23160-6 of São Paulo Research Foundation (FAPESP) and by the Grant 305361/2013-3 of National Council for Scientific and Technological Development (CNPq).

Appendix: Alternative weights in I-MTM

Other possible weight functions can be employed within MTM schemes without jeopardizing the ergodicity of the Markov chain. Let us consider the I-MTM scheme in Table 2 using a generic weight function $w_n(\mathbf{x})$, bounded and positive, i.e., $w_n(\mathbf{x}) > 0$, for all n. In this case, we have also to assume $\pi(\mathbf{x}) > 0$, for all $x \in \mathcal{X}$. As shown in Martino and Read (2013), Pandolfi et al. (2010), the adequate probability for accepting the jump from $\mathbf{x}_{t-1}$ to $\mathbf{z}_j$ in this case is

$$\alpha(\mathbf{x}_{t-1}, \mathbf{z}_j) = \min\left[1, \frac{\pi(\mathbf{z}_j)q_j(\mathbf{x}_{t-1})}{\pi(\mathbf{x}_{t-1})q_j(\mathbf{z}_j)}\frac{W_X}{W_Z}\right], \tag{14}$$

where

$$W_Z = \frac{w_j(\mathbf{z}_j)}{\sum_{n=1}^{N} w_n(\mathbf{z}_n)}, \qquad W_X = \frac{w_j(\mathbf{x}_{t-1})}{\left[\sum_{n=1}^{N} w_n(\mathbf{z}_n)\right] - w_j(\mathbf{z}_j) + w_j(\mathbf{x}_{t-1})}.$$

If the chosen weights are the importance weights, $w_n(\mathbf{x}) = \frac{\pi(\mathbf{x})}{q_n(\mathbf{x})}$, then Eq. (14) coincides with Eq. (12). Moreover, note that, in any case, $0 \leq W_Z \leq 1$ and $0 \leq W_X \leq 1$. As explained in Sect. 5, in general, it often occurs that $q_j(\mathbf{z}_j) > q_j(\mathbf{x}_{t-1})$ since $\mathbf{z}_j \sim q_j(\mathbf{z})$ whereas $\mathbf{x}_{t-1}$ has been generated from a generic q_k with $k \in \{1, \ldots, N\}$. Thus, $\frac{\pi(\mathbf{z}_j)q_j(\mathbf{x}_{t-1})}{\pi(\mathbf{x}_{t-1})q_j(\mathbf{z}_j)}$ tends to be close to zero and as consequence often $\alpha \approx 0$, regardless of the choice of the weight functions. Observe that if we employ the set of proposal pdfs $q_j(\mathbf{x})$'s as a mixture $\psi(\mathbf{x}) = \frac{1}{N}\sum_{n=1}^{N} q_n(\mathbf{x})$ as suggested in Sect. 5.1, the problem is solved also in this case.

References

Ali AM, Yao K, Collier TC, Taylor E, Blumstein D, Girod L (2007) An empirical study of collaborative acoustic source localization. In: Proceedings of information processing in sensor networks (IPSN07), Boston

Bédard M, Douc R, Mouline E (2012) Scaling analysis of multiple-try MCMC methods. Stoch Process Appl 122:758–786

Casarin R, Craiu R, Leisen F (2013) Interacting multiple try algorithms with different proposal distributions. Stat Comput 23(2):185–200

Cornuet JM, Marin JM, Mira A, Robert CP (2012) Adaptive multiple importance sampling. Scand J Stat 39(4):798–812

Craiu RV, Lemieux C (2007) Acceleration of the Multiple Try Metropolis algorithm using antithetic and stratified sampling. Stat Comput 17(2):109–120

Elvira V, Martino L, Luengo D, Bugallo M (2015a) Efficient multiple importance sampling estimators. IEEE Signal Process Lett 22(10):1757–1761

Elvira V, Martino L, Luengo D, Bugallo MF (2015b) Generalized multiple importance sampling. arXiv:1511.03095

Fitzgerald WJ (2001) Markov chain Monte Carlo methods with applications to signal processing. Signal Process 81(1):3–18

Frenkel D, Smit B (1996) Understanding molecular simulation: from algorithms to applications. Academic Press, San Diego

Hastings WK (1970) Monte Carlo sampling methods using Markov chains and their applications. Biometrika 57(1):97–109

Liang F, Liu C, Caroll R (2010) Advanced Markov chain Monte Carlo methods: learning from past samples. Wiley Series in Computational Statistics, England

Liu JS, Liang F, Wong WH (2000) The multiple-try method and local optimization in metropolis sampling. J Am Stat Assoc 95(449):121–134

Liu JS (2004) Monte Carlo strategies in scientific computing. Springer, Berlin

Martino L, Del Olmo VP, Read J (2012) A multi-point Metropolis scheme with generic weight functions. Stat Probab Lett 82(7):1445–1453

Martino L, Elvira V, Luengo D, Corander J (2015b) An adaptive population importance sampler: learning from the uncertanity. IEEE Trans Signal Process 63(16):4422–4437

Martino L, Elvira V, Luengo D, Corander J (2015c) Layered adaptive importance sampling. arXiv:1505.04732

Martino L, Elvira V, Luengo D, Corander J, Louzada F (2015a) Orthogonal parallel MCMC methods for sampling and optimization. arXiv:1507.08577

Martino L, Leisen F, and Corander J (2014) On Multiple Try schemes and the particle Metropolis-Hastings algorithm. arXiv:1409.0051

Martino L, Read J (2013) On the flexibility of the design of multiple try Metropolis schemes. Comput Stat 28(6):2797–2823

Metropolis N, Rosenbluth A, Rosenbluth M, Teller A, Teller E (1953) Equations of state calculations by fast computing machines. J Chem Phys 21:1087–1091

Owen A, Zhou Y (2000) Safe and effective importance sampling. J Am Stat Assoc 95(449):135–143

Pandolfi S, Bartolucci F, Friel N (2010) A generalization of the Multiple-try Metropolis algorithm for Bayesian estimation and model selection. J Mach Learn Res (workshop and conference proceedings volume 9: AISTATS 2010) 9:581–588

Qin ZS, Liu JS (2001) Multi-Point Metropolis method with application to hybrid Monte Carlo. J Comput Phys 172:827–840

Robert CP, Casella G (2004) Monte Carlo statistical methods. Springer, Berlin

Veach E, Guibas L (1995) Optimally combining sampling techniques for Monte Carlo rendering. In: SIGGRAPH 1995 proceedings, pp 419–428