Comput Stat (2013) 28:2797–2823
DOI 10.1007/s00180-013-0429-2

ORIGINAL PAPER

On the flexibility of the design of multiple try Metropolis schemes

Luca Martino · Jesse Read

Received: 27 December 2011 / Accepted: 25 June 2013 / Published online: 11 July 2013
© Springer-Verlag Berlin Heidelberg 2013

Abstract The multiple try Metropolis (MTM) method is a generalization of the classical Metropolis–Hastings algorithm in which the next state of the chain is chosen among a set of samples, according to normalized weights. In the literature, several extensions have been proposed. In this work, we show and remark upon the flexibility of the design of MTM-type methods, fulfilling the detailed balance condition. We discuss several possibilities, show different numerical simulations and discuss the implications of the results.

Keywords Metropolis–Hasting method · Multiple try Metropolis algorithm · Multi-point Metropolis algorithm · MCMC techniques

1 Introduction

Monte Carlo methods are very useful tools for scientific and approximate computing, numerical inference and optimization (Devroye 1986; Robert and Casella 2004). For instance, Monte Carlo methods are often necessary for the implementation of optimal Bayesian estimators for which several families of techniques have been proposed (Fitzgerald 2001; Gilks et al. 1995). The core of the Monte Carlo approach consists of drawing random samples from a target probability density function (pdf).

A very powerful class of Monte Carlo techniques is the so-called Markov Chain Monte Carlo (MCMC) algorithms (Gamerman and Lopes 2006; Gilks et al. 1995;

L. Martino (✉) · J. Read
Department of Signal Theory and Communications, Universidad Carlos III de Madrid,
Avenida de la Universidad 30, 28911 Leganés, Madrid, Spain
e-mail: luca@tsc.uc3m.es

J. Read
e-mail: jesse@tsc.uc3m.es

Liang et al. 2010; Liu 2004; Robert and Casella 2004). They generate a Markov chain such that its stationary distribution coincides with the target probability density function (pdf). Typically, the only requirement is to be able to evaluate the target function, where knowledge of the normalizing constant is usually not needed.

The most popular MCMC method is undoubtedly the Metropolis–Hasting (MH) algorithm (Hastings 1970; Metropolis et al. 1953). It can be applied to almost any arbitrary target distribution. However, to speed up the convergence and reduce the "burn-in" period, several extensions have been proposed in literature. For instance, the multiple try Metropolis (MTM) scheme (Liu et al. 2000) where, according to certain weights, the next state of the Markov chain is selected from a set of independent samples drawn from a generic proposal density. The main advantage of MTM is that it can explore a larger portion of the sample space without a drop in acceptance rate. Previously, a similar methodology was proposed in the domain of molecular simulation, called "orientational bias Monte Carlo" (Frenkel and Smit 1996, Chapter 13), where i.i.d. candidates are drawn from a *symmetric* proposal pdf and one of these is chosen according to normalized weights directly proportional to the target pdf.

Due to its good performance and the attractive possibility to combine it with adaptive MCMC strategies (Liang et al. 2010, Chapter 8; Haario et al. 2001) (for instance using different interacting or adaptive proposals at the same iteration Casarin et al. 2013), the basic formulation of the MTM has been modified and stressed in different ways. In (Pandolfi et al. 2010) the transition rule of the MTM algorithm is generalized such that the analytic form of the weights is not specified. They also study the extension of the MTM in the reversible jump framework. In (Casarin et al. 2013) a MTM scheme with different proposal is introduced. Different approaches with correlated candidates have been suggested in Craiu and Lemieux (2007), Martino et al. (2012a), Qin and Liu (2001). Some interesting theoretical results on the asymptotic behavior of different MTM strategies and some considerations on the choice of the weights are given in (Bédard et al. 2012).

In all the proposed MTM schemes the number of generated candidates is fixed in each iteration differently from, for instance, the delayed rejection Metropolis algorithm (Mira 2001; Tierney and Mira 1999) that generates sequentially different candidates until accepting one (or it is reached a maximum number of attempts). Furthermore, the state space is not augmented defining an extended target distribution, as in other MCMC methods based on auxiliary random variables (Storvik 2011).

In this work, we stress and remark upon the flexibility in the choice of transition rules within MTM algorithms. First of all, we mix the approaches from Casarin et al. (2013) and Pandolfi et al. (2010), building a MTM with generic weights using different proposal pdfs. Then, we present a general framework for the construction of acceptance probabilities in MTM schemes. We show this theoretically and illustrate with specific examples. Owing to this flexibility, it is also possible to design a MTM scheme without drawing reference points (Robert 2012). We also introduce this kind of MTM algorithm with a determinist reference points, and then discuss how this change affects its performance. Moreover, we show that all the presented schemes fulfill the detailed balance condition and provide numerical comparisons. Related considerations can be found in Barker (1965), Brooks (1998), Hastings (1970), Peskun (1973), Storvik (2011), Tierney (1994), Zhang and Zhang (2012).

The rest of the paper is organized as follows. In Sect. 2 we combine the schemes in (Casarin et al. 2013; Pandolfi et al. 2010) describing a MTM algorithm using different proposal densities and generic weight functions. In Sect. 3, we explain the flexibility in the choice of the acceptance functions, satisfying the detailed balance condition. Some examples of acceptance rules are shown in Sect. 4. Section 5 introduces a MTM method without generating the reference points randomly. Numerical comparisons are given in Sect. 6 and finally we draw conclusions in Sect. 7.

2 MTM algorithm with generic weights and different proposals

In the classical MH algorithm, a new possible state is drawn from the proposal pdf and the movement is accepted with a decision rule that guarantees fulfillment of the balance condition. In a multiple try approach, several (independent Liu et al. 2000; Pandolfi et al. 2010 or correlated Martino et al. 2012a; Qin and Liu 2001) samples are generated and from these a "good" one is chosen.

In (Casarin et al. 2013) the standard MTM is generalized using different proposal densities whereas in (Pandolfi et al. 2010) the authors extend the standard MTM considering generic weight functions. In the following section, we recall and mix together both approaches (Casarin et al. 2013; Pandolfi et al. 2010) providing an extended MTM algorithm drawing candidates from with different proposals where the weight functions are not defined specifically, i.e., the analytic form can be chosen arbitrarily (they must be bounded and positive functions).

2.1 Algorithm

Let $p_o(x)$ be the pdf that we want to draw from and $p(x)$ a function proportional to our target pdf $p_o(x)$ (i.e., $p(x) \propto p_o(x)$). Given a current state of the chain $x_t = x \in \mathcal{D} \subseteq \mathbb{R}$, $t \in \mathbb{N}$, (we assume scalar values only for simplicity in the treatment), we draw N independent samples each step from different proposal pdfs, i.e.,

$$y_1 \sim \pi_1(\cdot|x), y_2 \sim \pi_2(\cdot|x), \ldots, y_N \sim \pi_N(\cdot|x).$$

Therefore, we can write the joint distribution of the generated samples as

$$q_N(y_{1:N}|x) = \pi_1(y_1|x)\pi_2(y_2|x)\cdots\pi_N(y_N|x).$$

Then, a "good" candidate among the generated samples is chosen according to weight functions $\omega(z_1, z_2) \in \mathbb{R}^2 \to \mathbb{R}^+$ (where z_1 and z_2 are generic variables) that have to be (a) bounded and (b) positive. Given a current state $x_t = x$, the algorithm can be described as follows:

1. Draw N samples $y_{1:N} = [y_1, y_2, \ldots, y_N]$ from the joint pdf

$$q(y_{1:N}|x) = \pi_1(y_1|x)\pi_2(y_2|x)\pi_2(y_3|x)\cdots\pi_N(y_N|x),$$

namely, draw y_j from $\pi_j(\cdot|x)$, with $j = 1, \ldots, N$.

2. Calculate the weights $\omega_j(y_j, x)$, $j = 1, \ldots, N$, and normalize them to obtain $\bar{\omega}_j$, $j = 1, \ldots, N$.
3. Draw a $y = y_k \in \{y_1, \ldots, y_N\}$ according to $\bar{\omega}_j$, $j = 1, \ldots, N$ and set (recall that $y_k = y$)

$$W_y = \bar{\omega}_k = \frac{\omega_k(y, x)}{\sum_{j=1}^{N} \omega_j(y_j, x)}. \tag{1}$$

4. Draw other auxiliary samples (often called *reference points*),

$$x_i^* \sim \pi_i(\cdot|y)$$

for $i = 1, \ldots, k-1, k+1, \ldots, N$, and set $x_k^* = x$.
5. Compute the corresponding weights $\omega_j(x_j^*, y)$, $j = 1, \ldots, N$ and set (recall that $x_k^* = x$)

$$W_x = \frac{\omega_k(x, y)}{\sum_{j=1}^{N} \omega_j(x_j^*, y)}. \tag{2}$$

6. Let $x_{t+1} = y$ (recall that $y = y_k$) with probability

$$\alpha(x, y) = \min\left[1, \frac{p(y)\pi_k(x|y)}{p(x)\pi_k(y|x)} \frac{W_x}{W_y}\right], \tag{3}$$

otherwise set $x_{t+1} = x$ with the remaining probability $1 - \alpha(x, y)$.
7. Set $t = t + 1$ and go back to the step 1.

The kernel of the algorithm above satisfies the detailed balance condition. The proof is a special case of the development that we will present in Sect. 3.2, using the probability $\alpha(x, y)$ in Eq. (3).

2.2 Special case: standard MTM algorithm

Choosing the weight functions with the specific analytic form

$$\omega_i(y_i, x) = p(y_i)\pi_i(x|y_i)\lambda_i(x, y_i), \tag{4}$$

with $\lambda_i(x, y_i) = \lambda_i(y_i, x)$, $i = 1, \ldots, N$, we obtain the MTM scheme proposed in Casarin et al. (2013) (with different proposals). Indeed, note that the acceptance function (3) can be also expressed as

$$\alpha(x, y) = \min\left[1, \frac{p(y)\pi_k(x|y)}{p(x)\pi_k(y|x)} \frac{\omega_k(x, y)}{\omega_k(y, x)} \frac{\sum_{j=1}^{N} \omega_j(y_j, x)}{\sum_{j=1}^{N} \omega_j(x_j^*, y)}\right],$$

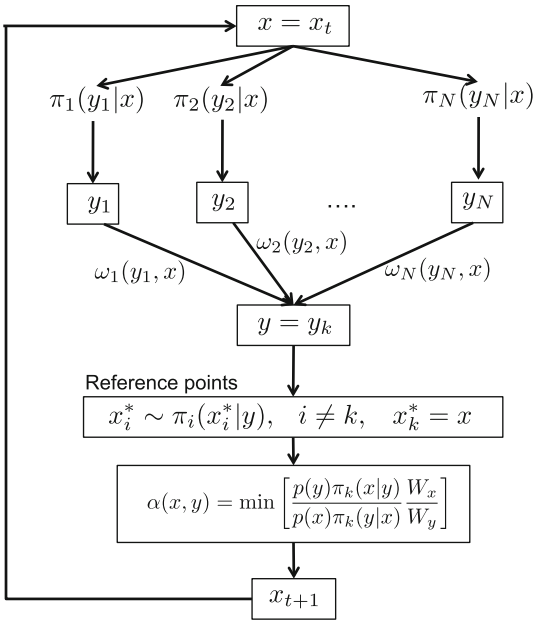

Fig. 1 Sketch of the MTM algorithm with generic weights and different proposals described in Sect. 2.1

and using the weight choice in Eq. (4),

$$\alpha(x, y) = \min \left[1, \frac{p(y)\pi_k(x|y)}{p(x)\pi_k(y|x)} \frac{p(x)\pi_k(y|x)\lambda_k(x, y)}{p(y)\pi_k(x|y)\lambda_k(y, x)} \frac{\sum_{j=1}^N \omega_j(y_j, x)}{\sum_{j=1}^N \omega_j(x_j^*, y)} \right],$$

then it is simplified

$$\alpha(x, y) = \min \left[1, \frac{\sum_{j=1}^N \omega_j(y_j, x)}{\sum_{j=1}^N \omega_j(x_j^*, y)} \right].$$

Finally, observe that if we use just one proposal, $\pi_1(y|x) = \pi_2(y|x) = \cdots = \pi_N(y|x)$ and the same functions $\lambda_1(x, y) = \lambda_2(x, y) = \cdots = \lambda_N(x, y)$, we obtain the standard formulation of the MTM (Liu et al. 2000). Figure 1 represents a general scheme of the algorithm described in Sect. 2.1.

2.3 Important observations

It is important to remark that, in order to obtain a fair comparison among the generated candidates, in the computation of the weights, it is advisable to use proposal functions with the same area below, i.e., $\int_{\mathcal{D}} \pi_1(y_1|x)dy_1 = \int_{\mathcal{D}} \pi_2(y_2|x)dy_2 = \cdots = \int_{\mathcal{D}} \pi_N(y_N|x)dy_N$, for instance they can be normalized. This is not strictly needed but recommendable.

Moreover, it is possible to show (see Sect. 3.2) that the algorithm above works owing to $\alpha(x, y)$ satisfies the following equation

$$p(x)\pi_k(y|x)W_y\alpha(x, y) = p(y)\pi_k(x|y)W_x\alpha(y, x). \tag{5}$$

Note that $0 \leq W_y \leq 1$ and $0 \leq W_x \leq 1$ are probabilities and functions of x, y, the remaining points y_i and x_i^*, then a more appropriate notation would be $W_y(y_1, \ldots, y_k = y, \ldots, y_N, x)$ and $W_x(x_1^*, \ldots x_k^* = x, \ldots, x_N^*, y)$.[1] However, for simplicity we maintain the notation W_y and W_x. In the sequel, we suggest different acceptance functions $\alpha(x, y)$.

3 Flexibility of the acceptance function

Here, we introduce different multiple try MH approaches with generic weights functions. Specifically we show how to design different suitable acceptance functions $\alpha(x, y)$ fulfilling the detailed balance condition. Indeed, it is possible to choose functions $\alpha(x, y)$ with the form

$$\alpha(x, y) = \beta(x, y)\gamma(x, y|\mathbf{x}_{-k}^*, \mathbf{y}_{-k}),$$

where

1. $\beta(x, y)$ is such that

$$p(x)\pi_k(y|x)\beta(x, y) = p(y)\pi_k(x|y)\beta(y, x), \quad \forall k \in \{1, \ldots, N\}, \tag{6}$$

2. $\gamma(x, y|\mathbf{x}_{-k}^*, \mathbf{y}_{-k})$ satisfies

$$W_y\gamma(x, y|\mathbf{x}_{-k}^*, \mathbf{y}_{-k}) = W_x\gamma(y, x|\mathbf{y}_{-k}, \mathbf{x}_{-k}^*), \tag{7}$$

   where $\mathbf{x}_{-k}^* = [x_1^*, \ldots x_{k-1}^*, x_{k+1}^*, \ldots, x_N^*]$ and $\mathbf{y}_{-k} = [y_1, \ldots y_{k-1}, y_{k+1}, \ldots, y_N]$.
3. Finally we need

$$0 \leq \alpha(x, y) \leq 1. \tag{8}$$

If the Eqs. (6) and (7) are jointly fulfilled then the condition (5) also holds, i.e., the equation

$$p(x)\pi_k(y|x)W_y\alpha(x, y) = p(y)\pi_k(x|y)W_x\alpha(y, x)$$

is satisfied. Equation (8) can be easily obtained choosing separately $0 \leq \beta(x, y) \leq 1$ and $0 \leq \gamma(x, y|\mathbf{x}_{-k}^*, \mathbf{y}_{-k}) \leq 1$. Moreover, in this case, Eq. (6) is exactly the balance condition of the standard MH algorithm, then we can choose any acceptance functions

[1] Recall that y_i are drawn from $\pi_i(\cdot|x)$ whereas x_i^* are drawn from $\pi_i(\cdot|y)$, $i = 1, \ldots, k-1, k+1, \ldots N$ and $x_k^* = x_t = x$.

suitable for the standard MH algorithm as function $\beta(x, y)$. Similar considerations can be used to design suitable functions $\gamma(x, y|\mathbf{x}^*_{-k}, \mathbf{y}_{-k})$. Some examples are provided in Sect. 4.

3.1 Algorithm

The novel scheme can be summarized as follows:

1. Draw N samples from the proposal pdfs $y_j \sim \pi_j(\cdot|x)$, with $j = 1, \ldots, N$.
2. Calculate the weights $\omega_j(y_j, x),\ j = 1, \ldots, N$, and normalize them to obtain $\bar{\omega}_j,\ j = 1, \ldots, N$.
3. Draw a $y = y_k \in \{y_1, \ldots., y_N\}$ according to $\bar{\omega}_j,\ j = 1, \ldots, N$ and set (recall that $y_k = y$)

$$W_y = \bar{\omega}_k = \frac{\omega_k(y, x)}{\sum_{j=1}^N \omega_j(y_j, x)}.$$

4. Draw other auxiliary samples $x_i^* \sim \pi_i(\cdot|y)$ for $i = 1, \ldots, k-1, k+1, \ldots, N$, and set $x_k^* = x$.
5. Compute the corresponding weights $\omega_j(x_j^*, y),\ j = 1, \ldots, N$ and set (recall that $x_k^* = x$)

$$W_x = \frac{\omega_k(x, y)}{\sum_{j=1}^N \omega_j(x_j^*, y)}.$$

6. Let $x_{t+1} = y$ (recall that $y = y_k$) with probability

$$\alpha(x, y) = \beta(x, y)\gamma(x, y|\mathbf{x}^*_{-k}, \mathbf{y}_{-k}),$$

where

$$p(x)\pi_k(y|x)\beta(x, y) = p(y)\pi_k(x|y)\beta(y, x)$$

and

$$W_y\gamma(x, y|\mathbf{x}^*_{-k}, \mathbf{y}_{-k}) = W_x\gamma(y, x|\mathbf{y}_{-k}, \mathbf{x}^*_{-k}).$$

Otherwise set $x_{t+1} = x$ with the remaining probability $1 - \alpha(x, y)$.
7. Set $t = t + 1$ and go back to the step 1.

3.2 Balance condition

To guarantee that a Markov chain generated by an MCMC method converges to the target distribution $p_o(x) \propto p(x)$, we can prove that the kernel $A(y|x)$ of the corresponding algorithm (probability of accepting a generated sample y given the previous

state value x) fulfills the following detailed balance condition[2] (Liu 2004; Robert and Casella 2004)

$$p(x)A(y|x) = p(y)A(x|y).$$

First of all, we need to write down the kernel $A(y|x)$. We consider $x \neq y$, since the case $x = y$ is trivial (indeed, in this case $A(y|x)$ is proportional to a delta function $\delta(y-x)$ (Liu 2004; Robert and Casella 2004)). The kernel (for $x \neq y$) can be expressed as

$$A(y = y_k|x) = \sum_{i=1}^{N} h(y = y_k|x, k = i),$$

where $h(y = y_k|x, k = i)$ is the probability of accepting the new state $x_{t+1} = y_k$ given the previous one $x_t = x$, when the chosen sample y_k is the ith candidate, i.e., when $y_k = y_i$. However, since the y_i are exchangeable, for symmetry we have $h(y = y_k|x, i) = h(y = y_k|x, j)\, \forall i, j \in \{1, \ldots, N\}$. Hence, we can also write

$$A(y = y_k|x) = N \cdot h(y = y_k|x, k),$$

where $k \in \{1, \ldots, N\}$ and we recall N is the total number of proposed candidates y_i. Then, we need to show that

$$p(x)h(y|x, k) = p(y)h(x|y, k),$$

for a generic $k \in \{1, \ldots, N\}$. Following each step of the algorithm above, we can write

$$\begin{aligned} &p(x)h(y = y_k|x, k) \\ &= p(x) \int_{\mathcal{D}} \cdots \int_{\mathcal{D}} \left[\prod_{j=1}^{N} \pi_j(y_j|x)\right] \frac{\omega_k(y, x)}{\sum_{i=1}^{N} \omega_i(y_i, x)} \left[\prod_{j=1; j\neq k}^{N} \pi_j(x_j^*|y)\right] \cdot \\ &\times \underbrace{\beta(x, y)\gamma(x, y|\mathbf{x}_{-k}^*, \mathbf{y}_{-k})}_{\alpha(x,y)}\, dy_{1:k-1} dy_{k+1:N} dx_{1:k-1}^* dx_{k+1:N}^*. \end{aligned}$$

Note that each factor inside the integral corresponds to a step of the method described in the previous section. The integral is over all auxiliary variables. Since we consider $y = y_k$ and recalling the definition of W_y in Eq. (1), we can rewrite the expression in this way

[2] Note that the balance condition is a sufficient but not necessary condition. Namely, the detailed balance ensures invariance. The converse is not true. Markov chains that satisfy the detailed balance condition are called *reversible*.

$$
\begin{aligned}
&p(x)h(y|x,k)\\
&= p(x)\int_{\mathcal{D}}\cdots\int_{\mathcal{D}}\pi_k(y|x)\left[\prod_{j=1,j\neq k}^{N}\pi_j(y_j|x)\right]W_y\left[\prod_{j=1;j\neq k}^{N}\pi_j(x_j^*|y)\right]\\
&\quad\times\beta(x,y)\gamma(x,y|\mathbf{x}^*_{-k},\mathbf{y}_{-k})\,dy_{1:k-1}dy_{k+1:N}dx^*_{1:k-1}dx^*_{k+1:N}.
\end{aligned}
$$

and we only arrange it, obtaining

$$
\begin{aligned}
&p(x)h(y|x,k)\\
&= \int_{\mathcal{D}}\cdots\int_{\mathcal{D}}\left[\prod_{j=1,j\neq k}^{N}\pi_j(y_j|x)\right]\left[\prod_{j=1;j\neq k}^{N}\pi_j(x_j^*|y)\right]\\
&\quad\times p(x)\pi_k(y|x)\beta(x,y)\times W_y\gamma(x,y|\mathbf{x}^*_{-k},\mathbf{y}_{-k})\,d\mathbf{y}_{-k}d\mathbf{x}^*{}_{-k}. \qquad (9)
\end{aligned}
$$

Therefore, since we assume (see Eqs. 6 and 7)

$$p(x)\pi_k(y|x)\beta(x,y) = p(y)\pi_k(x|y)\beta(y,x),$$

and

$$W_y\gamma(x,y|\mathbf{x}^*_{-k},\mathbf{y}_{-k}) = W_x\gamma(y,x|\mathbf{y}_{-k},\mathbf{x}^*_{-k}),$$

it is straightforward that the expression in Eq. (9) is symmetric in x and y. Indeed, we can exchange the notations of x and y, and x_i^* and y_j, respectively, and the expression does not vary. Then we can write

$$p(x)h(y|x,k) = p(y)h(x|y,k).$$

Since we have assumed a generic k and $A(y = y_k|x) = h(y = y_k|x,k)$, it possible to assert that

$$p(x)A(y|x) = p(y)A(x|y),$$

that is the balance condition. Therefore, the Markov chain generated by the algorithm, described in the previous section, converges to our target pdf.

4 Examples of functions $\alpha(x, y)$

In this section, we provide some suitable acceptance functions $\alpha(x,y) = \mathcal{D}\times\mathcal{D}\to[0,1]$, that satisfies the condition (5). The easiest way is to obtain $\alpha(x,y)$ is to design separately suitable functions $0\le\beta(x,y)\le 1$ and $0\le\gamma(x,y|\mathbf{x}^*_{-k},\mathbf{y}_{-k})\le 1$.

Table 1 Example of suitable functions $\beta(x, y)$

Functions $\beta(x, y)$	References
$\beta_1(x, y) = \min\left[1, \frac{p(y)\pi_k(x\|y)}{p(x)\pi_k(y\|x)}\right]$	Hastings (1970), Metropolis et al. (1953)
$\beta_2(x, y) = \frac{p(y)\pi_k(x\|y)}{p(x)\pi_k(y\|x)+p(y)\pi_k(x\|y)}$	Barker (1965)
$\beta_3(x, y) = \frac{\lambda(x,y)}{1+\frac{p(x)\pi_k(y\|x)}{p(y)\pi_k(x\|y)}}$	Hastings (1970)
$\beta_4(x, y) = \frac{p(y)\pi_k(x\|y)}{\lambda(x,y)}$	Liu (2004), Robert and Casella (2004)
$\beta_5(x, y) = \frac{\lambda(x,y)}{p(x)\pi_k(y\|x)}$	Liu (2004), Robert and Casella (2004)
$\beta_6(x, y) = \frac{p(y)\lambda(x,y)}{\pi_k(y\|x)}$	Liu (2004, Chapter 5)
$\beta_7(x, y) = \frac{\pi_k(x\|y)\lambda(x,y)}{p(x)}$	Liu (2004, Chapter 5)

4.1 Possible choices of $\beta(x, y)$

To design a function $\beta(x, y)$ such that $0 \leq \beta(x, y) \leq 1$ and

$$p(x)\pi_k(y|x)\beta(x, y) = p(y)\pi_k(x|y)\beta(y, x),$$

we can choose any acceptance rule suitable for the standard MH algorithm (Barker 1965; Hastings 1970). Hence, for instance, we can choose the classical acceptance rule of the MH algorithm, i.e.,

$$\beta_1(x, y) = \min\left[1, \frac{p(y)\pi_k(x|y)}{p(x)\pi_k(y|x)}\right]. \tag{10}$$

Other possibilities are summarized in Table 1 where $\lambda(x, y)$ is a symmetric non-negative function (i.e., $\lambda(x, y) \geq 0$ and $\lambda(x, y) = \lambda(y, x)$ for all $(x, y) \in \mathcal{D} \times \mathcal{D}$) such that $0 \leq \beta(x, y) \leq 1$.

Moreover, defining

$$R(x, y) = \frac{p(y)\pi_k(x|y)}{p(x)\pi_k(y|x)},$$

and considering a function $F(\vartheta) : \mathbb{R}^+ \to [0, 1]$ such that

$$F(\vartheta) = \vartheta F(1/\vartheta),$$

then it is possible to define a general acceptance function (Gamerman and Lopes 2006; Gilks et al. 1995)

$$\beta_g(x, y) = (F \circ R)(x, y) = F(R(x, y)).$$

For instance, if $F(\vartheta) = \min[1, \vartheta]$ we obtain Eq. (10) and if $F(\vartheta) = \frac{\vartheta}{1+\vartheta}$ we find β_2 or β_3 with $\lambda(x, y) = 1$ (see Table 1). In Peskun (1973) there is a comparison of different acceptance functions in a standard MH algorithm.

4.2 Possible choices of $\gamma(x, y|\mathbf{x}^*_{-k}, \mathbf{y}_{-k})$

In this section, we provide some examples of suitable function $\gamma(x, y|\mathbf{x}^*_{-k}, \mathbf{y}_{-k})$. We need functions $\gamma(x, y|\mathbf{x}^*_{-k}, \mathbf{y}_{-k})$ such that

$$W_y \gamma(x, y|\mathbf{x}^*_{-k}, \mathbf{y}_{-k}) = W_x \gamma(y, x|\mathbf{y}_{-k}, \mathbf{x}^*_{-k}), \tag{11}$$

where

$$W_y = \frac{\omega_k(y, x)}{\sum_{j=1}^N \omega_j(y_j, x)}, \quad \text{and} \quad W_x = \frac{\omega_k(x, y)}{\sum_{j=1}^N \omega_j(x^*_j, y)}.$$

Therefore, for instance, it is possible to choose

$$\gamma_1(x, y|\mathbf{x}^*_{-k}, \mathbf{y}_{-k}) = W_x.$$

Indeed, in this case $\gamma(y, x|\mathbf{y}_{-k}, \mathbf{x}^*_{-k}) = W_y$ and the condition (11) is satisfied ($W_y W_x = W_x W_y$). Another possibility is to define

$$\gamma_2(x, y|\mathbf{x}^*_{-k}, \mathbf{y}_{-k}) = \frac{W_x}{W_x + W_y},$$

or

$$\gamma_3(x, y|\mathbf{x}^*_{-k}, \mathbf{y}_{-k}) = \min\left[1, \frac{W_x}{W_y}\right].$$

5 MTM without drawing reference points

The previous considerations also suggest how it is possible to design a MTM that avoids sampling the reference points $\mathbf{x}^*_{-k}$. For some authors generating the reference samples is considered a drawback of the MTM schemes, since $N - 1$ samples are *only* drawn to fulfill the balance condition (Robert 2012). To avoid this step, the MTM method in Sect. 2.1 can be modified as follows:

1. Given a current state $x_t = x$, draw N samples $y_{1:N} = [y_1, y_2, \ldots, y_N]$ from the joint pdf

$$q(y_{1:N}|x) = \pi_1(y_1|x)\pi_2(y_2|x)\pi_2(y_3|x)\cdots\pi_N(y_N|x),$$

namely, draw y_j from $\pi_j(\cdot|x)$, with $j = 1, \ldots, N$.

2. Calculate the weights $\omega_j(y_j, x)$, $j = 1, \ldots, N$, and normalize them to obtain $\bar{\omega}_j$, $j = 1, \ldots, N$.
3. Draw a $y = y_k \in \{y_1, \ldots., y_N\}$ according to $\bar{\omega}_j$, $j = 1, \ldots, N$ and set

$$W_y = \bar{\omega}_k = \frac{\omega_k(y, x)}{\sum_{j=1}^{N} \omega_j(y_j, x)}. \tag{12}$$

4. Set $x_i^* = y_i$ for $i = 1, \ldots, k-1, k+1, \ldots., N$, and set $x_k^* = x$.
5. Compute the corresponding weights $\omega_j(x_j^*, y)$, $j = 1, \ldots, N$ and (recalling $x_k* = x$) set

$$W_x = \frac{\omega_k(x, y)}{\sum_{j=1}^{N} \omega_j(x_j^*, y)}. \tag{13}$$

6. Let $x_{t+1} = y$ (recall that $y = y_k$) with probability

$$\alpha(x, y) = \min\left[1, \frac{p(y)\prod_{i=1}^{N} \pi_i(x_i^*|y)}{p(x)\prod_{i=1}^{N} \pi_i(y_i|x)} \frac{W_x}{W_y}\right], \tag{14}$$

   otherwise set $x_{t+1} = x$ with the remaining probability $1 - \alpha(x, y)$.
7. Set $t = t + 1$ and go back to the step 1.

The differences w.r.t. the standard MTM method are contained in the steps 4 and 6. In this case the vectors $\mathbf{y} = [y_1, \ldots, y_k = y, \ldots., y_N]$ and $\mathbf{x}^* = [x_1^* = y_1, \ldots, x_k^* = x, \ldots., x_N^* = y_N]$ differ only in the position k, i.e., $\mathbf{y}_{-k} = \mathbf{x}_{-k}^*$. Hence, note that $\alpha(x, y)$ can be expressed as

$$\alpha(x, y) = \min\left[1, \frac{p(y)\pi_k(x|y)}{p(x)\pi_k(y|x)} \frac{\prod_{i\neq k}^{N} \pi_i(y_i|y)}{\prod_{i\neq k}^{N} \pi_i(y_i|x)} \frac{W_x}{W_y}\right]. \tag{15}$$

However, although this scheme satisfies the balance condition as we show below, observing the expression of α, a drawback could seem evident: since the samples $y_{1:N}$ are drawn from $\pi_i(\cdot|x)$, $i = 1, \ldots, N$, the product $\prod_{i\neq k}^{N} \pi_i(y_i|x)$ would be "often" greater then $\prod_{i\neq k}^{N} \pi_i(y_i|y)$. That is to say, x is more "likely" than y given the "observations" y_i, $i \neq k$. Therefore, $\alpha(x, y)$ would be "often" less than 1 so that accepting a jump becomes "rare."[3] This issue would increase with $N \to +\infty$. However, the numerical simulations (see Sect. 6) show that the probability $\alpha(x, y)$ first surprisingly increases for small values of N (owing to the factor $\frac{W_x}{W_y}$) and then decreases with $N \to +\infty$ as expected. Moreover the performance generally gets worse with $N \to +\infty$. Hence this scheme appears, in general, useless. These considerations above explain as, in the standard MTM version

[3] However, it is important to remark that high acceptance rates are not a suitable indicator of good performance since, in general, the best acceptance rate is different from 1 (Roberts et al. 1997).

(Liu et al. 2000), the authors introduce the idea of randomly generating the reference points x_i^*. However, there is an important special case that we show in Sect. 5.2.

5.1 Balance condition

Again we must check that the detailed balance condition $p(x)A(y|x) = p(y)A(x|y)$ is fulfilled. The kernel $A(y|x)$ (for $x \neq y$) can be expressed, also in this case, as $A(y = y_k|x) = N \cdot h(y = y_k|x, k)$, where $k \in \{1, \ldots, N\}$ and N is the total number of proposed candidates y_i. Then, finally we have to show that

$$p(x)h(y|x, k) = p(y)h(x|y, k),$$

for a generic $k \in \{1, \ldots, N\}$. Following each step of the MTM algorithm without reference point, we can write

$$\begin{aligned} p(x)h(y|x, k) = p(x) \int_{\mathcal{D}} \cdots \int_{\mathcal{D}} \left[\prod_{i=1}^{N} \pi_i(y_i|x) \right] W_y \\ \min \left[1, \frac{p(y) \prod_{i=1}^{N} \pi_i(x_i^*|y)}{p(x) \prod_{i=1}^{N} \pi_i(y_i|x)} \frac{W_x}{W_y} \right] \\ dy_{1:k-1} dy_{k+1:N} dx_{1:k-1}^* dx_{k+1:N}^*. \end{aligned}$$

The integral is over all auxiliary variables. Just by rearranging, we obtain

$$\begin{aligned} p(x)h(y|x, k) = \int_{\mathcal{D}} \cdots \int_{\mathcal{D}} \\ \min \left[p(x) \prod_{i=1}^{N} \pi_i(y_i|x) W_y, \, p(y) \prod_{i=1}^{N} \pi_i(x_i^*|y) W_x \right] \\ dy_{1:k-1} dy_{k+1:N} dx_{1:k-1}^* dx_{k+1:N}^*. \end{aligned} \tag{16}$$

Recalling that $x_j^* = y_j$ for $j = 1, \ldots, k-1, k+1, \ldots, N$, $x_k^* = x$ and $y_k = y$, the Eq. (16) can be rewritten as

$$\begin{aligned} p(x)h(y|x, k) = \int_{\mathcal{D}} \cdots \int_{\mathcal{D}} \\ \min \left[p(x)\pi_k(y|x) \prod_{i \neq k}^{N} \pi_i(y_i|x) W_y, \, p(y)\pi_k(x|y) \prod_{i \neq k}^{N} \pi_i(y_i|y) W_x \right] \\ dy_{1:k-1} dy_{k+1:N}. \end{aligned}$$

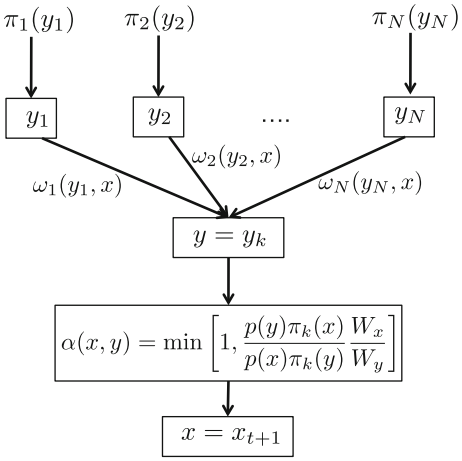

Fig. 2 Scheme of MTM algorithm with generic weights and different independent proposal pdfs

Therefore it is straightforward to see that we can exchange x and y without varying the expression above (see also Eq. 12 and 13), then $p(x)h(y|x,k) = p(y)h(x|y,k)$ and the balance condition $p(x)A(y|x) = p(y)A(x|y)$ is satisfied.

5.2 Independent proposal pdfs

If the proposal pdfs are chosen as independent densities, i.e., $\pi_1(y_1|x) = \pi_1(y_1)$, $\pi_2(y_2|x) = \pi_2(y_2) \ldots \pi_N(y_N|x) = \pi_N(y_N)$, the algorithm is simplified. Indeed, the $\alpha(x, y)$ probability in Eq. (15), i.e.,

$$\alpha(x, y) = \min\left[1, \frac{p(y)\pi_k(x|y)}{p(x)\pi_k(y|x)}\frac{\prod_{i\neq k}^{N}\pi_i(y_i|y)}{\prod_{i\neq k}^{N}\pi_i(y_i|x)}\frac{W_x}{W_y}\right],$$

now it can be rewritten as

$$\begin{aligned}\alpha(x, y) &= \min\left[1, \frac{p(y)\pi_k(x)}{p(x)\pi_k(y)}\frac{\prod_{i\neq k}^{N}\pi_i(y_i)}{\prod_{i\neq k}^{N}\pi_i(y_i)}\frac{W_x}{W_y}\right] \\ &= \min\left[1, \frac{p(y)\pi_k(x)}{p(x)\pi_k(y)}\frac{W_x}{W_y}\right].\end{aligned}$$

Observe that it is exactly the probability $\alpha(x, y)$ obtained in Eq. (3) using independent proposals. Therefore, here, the conclusion is different from the general case: it is not necessary to draw reference points when independent proposal densities are used. It is necessary just to set deterministically $x_i^* = y_i$ for $i = 1, \ldots, k-1, k+1, \ldots, N$, and set $x_k^* = x$. This special case, when the weights are chosen as in Sect. 2.2, is also discussed in Liu (2004, Chapter 5).

Figure 2 depicts the scheme of a MTM with generic weights and different independent proposal pdfs, whereas Fig. 3 shows virtually the simplest MTM algorithms, using

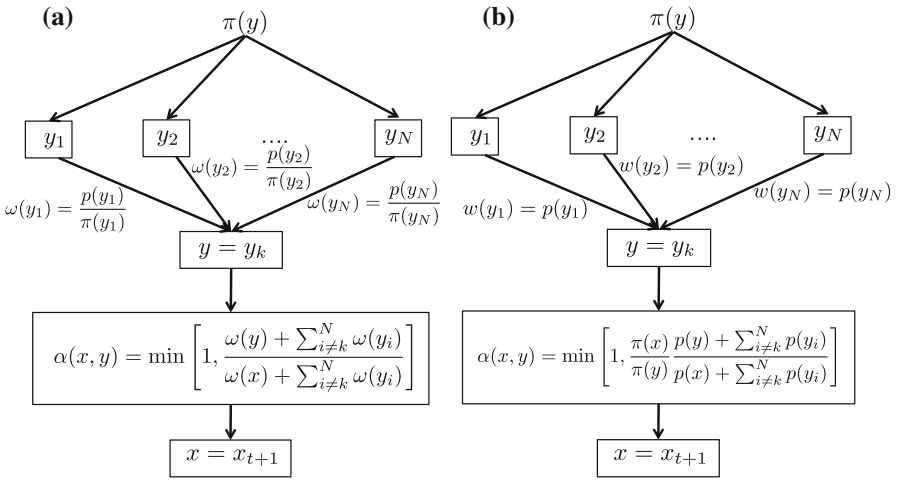

Fig. 3 Sketch of the simplest MTM schemes using just one independent proposal density, **a** with importance weights and **b** weights proportional to $p(x)$

the same independent proposal to draw the N candidates and importance weights (Fig. 3a) or weights proportional to the target (Fig. 3b).[4] In this special cases, the analysis of the algorithm is also simpler. Indeed, for instance, consider the case in Fig. 3a. The acceptance probability can be expressed as

$$\alpha(x, y) = \min\left[1, \frac{\omega(y) + \sum_{i\neq k}^{N} \omega(y_i)}{\omega(x) + \sum_{i\neq k}^{N} \omega(y_i)}\right],$$

where $w(y_i) = \frac{p(y_i)}{\pi(y_i)}$. Note that, in this case clearly $\alpha(x, y) \to 1$ as $N \to \infty$, since the chosen candidate is "extremely good" using the importance sampling principle, when $N \to \infty$.

6 Numerical simulations

In this section, we provide numerical results comparing different MTM approaches: using random walks or independent proposal pdfs, with different weight functions, without drawing the reference points and using different acceptance functions. All the results have been averaged over 2,000 runs and they are obtained generating 5,000 iterations of the Markov chain, with the exception of the last example where we only draw 500 samples.

[4] Another simple MTM scheme is the "orientational bias Monte Carlo" (Frenkel and Smit 1996, Chapter 13). In this case, the proposal pdf must be symmetric, i.e., $\pi(y|x) = \pi(x|y)$, and the weights must be proportional to the target, i.e., $\omega(y_i) = p(y_i)$, $i = 1, \ldots, N$.

Table 2 Numerical results (proposal as random walk, $\sigma = 2$, using importance weights)

Technique	Number of tries	Acceptance rate	Linear correlation
standard MH (MTM with $N = 1$)	$N = 1$	0.3002	0.9053
MTM-rw	$N = 2$	0.4363	0.8397
MTM-rw	$N = 5$	0.6046	0.6989
MTM-rw	$N = 100$	0.8647	0.1892
MTM-rw	$N = 1{,}000$	0.9557	0.0513
MTM-without	$N = 2$	0.4229	0.9160
MTM-without	$N = 5$	0.5121	0.9568
MTM-without	$N = 100$	0.1902	0.9978
MTM-without	$N = 1{,}000$	0.0036	0.9993

6.1 Random walk proposal densities

Let $X \in \mathbb{R}$ be a random variable[5] with bimodal pdf

$$p_o(x) \propto p(x) = \exp\left\{-(x^2-4)^2/4\right\} = \exp\left\{-\frac{x^4 - 8x^2 + 16}{4}\right\}. \tag{17}$$

We want to draw samples from $p_o(x)$ using different MTM schemes. We generate tries from a Gaussian proposal with variance σ^2 and the mean depends on the previous state x of the chain, i.e.,

$$\pi(y|x) \propto \exp\left\{-\frac{(y-x)^2}{2\sigma^2}\right\}. \tag{18}$$

We apply MTM methods using the proposal above, different number of candidates $N = 1, 2, 5, 100, 1{,}000$ and different standard deviation $\sigma = 2, 10$. Importance weights $\omega(y_i, x) = \frac{p(y_i)}{\pi(y_i|x)}$ are used to select a good candidate. Observe that a MTM with $N = 1$ is exactly a standard MH algorithm. We also apply different MTM techniques without drawing the reference points (denoted as "MTM-without") described in Sect. 5. Tables 2 and 3 summarize the numerical results in terms of averaged probability of accepting a movement and linear correlation between the state x_t and x_{t+1}.

It is important to remark that high acceptance rates are not a suitable indicator of good performance since, in general, the best acceptance rate is different from 1 (Roberts et al. 1997). Therefore, better performance is indicated by smaller correlations. We show also the acceptance rates because of the MTM method (drawing the reference

[5] Note that, in this work, we have mainly considered scalar variables in order to simplify the treatment and the notation. All the considerations and algorithms contained in this work are also valid for multi-dimensional variables (see, for instance, the last numerical example in Sect. 6.6).

Table 3 Numerical results (proposal as random walk, $\sigma = 10$, using importance weights)

Technique	Number of tries	Acceptance rate	Linear correlation
standard MH (MTM with $N = 1$)	$N = 1$	0.0991	0.9085
MTM-rw	$N = 2$	0.1795	0.8335
MTM-rw	$N = 5$	0.3483	0.6700
MTM-rw	$N = 100$	0.8373	0.1676
MTM-rw	$N = 1{,}000$	0.9483	0.0522
MTM-without	$N = 2$	0.1810	0.8376
MTM-without	$N = 5$	0.3575	0.7017
MTM-without	$N = 100$	0.4453	0.9264
MTM-without	$N = 1{,}000$	0.2612	0.9952

points) presents a behavior typical in adaptive MCMC algorithms where the adaptive proposal pdf convergence to the true shape of the target (Martino et al. 2012b): the acceptance rate grows and the linear correlation decreases quickly as $N \to +\infty$. Indeed, we can observe that, in both cases $\sigma = 2, 10$, the correlation obtained with the MTM decreases to zero as $N \to +\infty$. Without drawing the reference points, the resulting algorithm is totally useless for $\sigma = 2$ (Table 2) whereas it outperforms the standard MH for $N = 2$ and $N = 5$ for $\sigma = 10$ (Table 3). However, increasing N the performance gets worse. The results in Table 3 suggest that it exists an *optimal* number of tries for a MTM scheme without generating randomly the reference points. However, the MTM method with the additional cost of the random generation of reference points always outperforms the general scheme described in Sect. 5. With independent proposal pdfs this is not true as we show later.

6.2 Different choice of the weights

Considering the same target pdf in Eq. (17), the Gaussian proposal with $\sigma = 10$ in Eq. (18) (random walk) and using $N = 100$ tries, we have compared the performance of different weight functions. Table 4 summarizes the results.

The best results are provided by the importance weights $\omega_i(y_i, x) = \frac{p(y_i)}{\pi_i(y_i|x)}$. The weights of the form $\omega_i(y_i, x) = p(y_i)$ and $\omega_i(y_i, x) = p(y_i)\pi_i(x|y_i)$ also yield small correlation. Clearly, the choice $\omega_i(y_i, x) = 1$ produces the same results of a standard MH since the selected candidate is chosen uniformly among the set of drawn tries y_i, $i = 1, \ldots, N$, without using any information of the target or the proposal functions.

6.3 Independent proposal densities

In order to draw samples from the target in Eq. (17), we also apply MTM algorithms with independent proposal densities (MTM-ind) as

Table 4 Numerical results (proposal as random walk, $\sigma = 10,\ N = 100$ tries)

Weights	Acceptance rate	Linear correlation
$\omega_i(y_i, x) = \frac{p(y_i)}{\pi_i(y_i\|x)}$ importance weights	0.8373	0.1676
$\omega_i(y_i, x) = p(y_i)$	0.8374	0.1959
$\omega_i(y_i, x) = 1$	0.0988	0.9090
$\omega_i(y_i, x) = \sqrt{p(y_i)}$	0.7036	0.3340
$\omega_i(y_i, x) = [p(y_i)]^2$	0.6870	0.3093
$\omega_i(y_i, x) = [p(y_i)]^3$	0.4476	0.4020
$\omega_i(y_i, x) = \pi_i(x\|y_i)$	0.1348	0.8809
$\omega_i(y_i, x) = \frac{1}{\pi_i(y_i\|x)}$	0.0365	0.9652
$\omega_i(y_i, x) = p(y_i)\pi_i(x\|y_i)$	0.8371	0.2248

Table 5 Numerical results ($\sigma = 10,\ N = 100$ tries)

Proposal pdfs	Acceptance rate	Linear correlation
MTM-rw with $\omega_i(y_i, x) = \frac{p(y_i)}{\pi_i(y_i\|x)}$	0.8373	0.1676
MTM-rw with $\omega_i(y_i, x) = p(y_i)$	0.8374	0.1959
MTM-ind with one proposal pdf ($\mu = 0$) and $\omega_i(y_i, x) = \frac{p(y_i)}{\pi_i(y_i\|x)}$	0.9760	0.0252
MTM-ind with one proposal pdf ($\mu = 0$) and $\omega_i(y_i, x) = p(y_i)$	0.9751	0.0267
MTM-ind with two proposal pdfs ($\mu_1 = -10$ and $\mu_2 = 2$) and $\omega_i(y_i, x) = \frac{p(y_i)}{\pi_i(y_i\|x)}$	0.7420	0.2748
MTM-ind with two proposal pdfs ($\mu_1 = -10$ and $\mu_2 = 2$) and $\omega_i(y_i, x) = p(y_i)$	0.7509	0.6622

$$\pi(y) \propto \exp\left\{-\frac{(y-\mu)^2}{2\sigma^2}\right\},$$

with $\sigma = 10$. In a first scheme, we generate $N = 100$ candidates from one proposal with $\mu = 0$. Moreover, in other scheme, we use two different independent proposal pdfs with $\mu_1 = -10$ and $\mu_2 = 2$. In this case, we draw $N/2 = 50$ tries from each one. We apply these schemes with importance weights, $\omega_i(y_i, x) = \frac{p(y_i)}{\pi_i(y_i)}$, and also with weights just proportional to the target pdf, $\omega_i(y_i, x) = p(y_i)$. Table 5 shows the numerical results.

The first two lines of the Table 5 recall the acceptance rates and the linear correlations using the random walk proposal densities. The table shows that the MTM with independent proposal with $\mu = 0$ provides the best results, i.e., the smallest correlation. However, the results depend strongly on a suitable tuning of the parameter μ. Also in this case, the importance weights seem to provide

Table 6 Estimation of the constant $\frac{1}{c_p} = \sqrt{\frac{2}{2\pi}} = 0.5642$ and standard deviation of the estimation ($N = 1000$ tries)

Technique	Estimation of $\frac{1}{c_p}$	Std of the estimation	Further informations
MTM-ind	0.6056	0.0012	$\mu = 10,\ \sigma = 50$
MTM-ind	0.5994	0.0010	$\mu = 100,\ \sigma = 50$
MTM-rw	0.5819	0.0050	$\sigma = 50$

better results. Another important consideration is that, using two proposal pdfs, the MTM has selected a candidate generated from the proposal with $\mu_1 = -10$ with a rate of 39.5 % using importance weights, and just 1.5 % with the weights proportional to the target. This observation can be extremely important to design an adaptive strategy where the best proposal density is chosen among of a set of proposals.

6.4 Heavy tails

In order to analyze the performance of the MTM schemes with heavy tails, now we consider as target pdf the so-called *Lévy distribution* for non-negative random variable, namely,

$$p_o(x) \propto p(x) = \frac{1}{(x-\eta)^{3/2}} \exp\left(-\frac{\nu}{2(x-\eta)}\right), \quad \forall x \geq \eta \geq 0. \tag{19}$$

The normalizing constant $\frac{1}{c_p}$, such that $p_o(x) = \frac{1}{c_p} p(x)$, is analytically known, $\frac{1}{c_p} = \sqrt{\frac{\nu}{2\pi}}$.

Moreover, given a random variable $X \sim p_o(x)$, all the moments $E[X^\gamma]$ with $\gamma \geq 1$ do not exist owing to the heavy tail characteristic of the Lévy distribution.

Our goal is to estimate the normalizing constant $\frac{1}{c_p}$ via Monte Carlo simulation, when $\eta = 0$ and $\nu = 2$, generating 5,000 iterations of the Markov chain. We apply three different MTM techniques with $N = 1{,}000$ tries (drawing the reference points) and using importance weights to choose a suitable candidate each step. In the first two schemes (MTM-ind), we use an independent proposal $\pi(x_t) \propto \exp\{-(x_t - \mu)^2/(2\sigma^2)\}$ with $\mu = 10, 100$ and $\sigma = 50$, whereas, in the last one (MTM-rw), we use a random walk proposal $\pi(x_t|x_{t-1}) \propto \exp\{-(x_t - x_{t-1})^2/(2\sigma^2)\}$ with $\sigma = 50$. We choose huge values of σ due to the heavy tail feature of the target. We have averaged all the results over 2,000 runs and they are summarized in Table 6. The real value of $\frac{1}{c_p}$ when $\nu = 2$ is $\sqrt{\frac{2}{2\pi}} = 0.5642$.[6]

[6] We do not provide the estimated linear correlation because of the moments (as the mean, for instance) of the target do not exist, and it makes difficult a right estimation of the correlation.

Table 7 Numerical results with $N = 10$

Function α	Acceptance rate	Linear correlation
$\alpha_{1,1}(x, y)$	0.1167	0.9932
$\alpha_{1,2}(x, y)$	0.3246	0.9811
$\alpha_{1,3}(x, y)$	0.5512	0.9756
$\alpha_{2,3}(x, y)$	0.3370	0.9806

Table 8 Numerical results with $N = 100$

Function α	Acceptance rate	Linear correlation
$\alpha_{1,1}(x, y)$	0.0173	0.9931
$\alpha_{1,2}(x, y)$	0.3354	0.9828
$\alpha_{1,3}(x, y)$	0.5904	0.9737
$\alpha_{2,3}(x, y)$	0.3540	0.9859

6.5 Different acceptance probabilities

In this section, we consider again the bimodal target density in Eq. (17), i.e., $p_o(x) \propto p(x) = \exp\left\{-(x^2-4)^2/4\right\}$, and we generate candidates from a random walk Gaussian density with $\sigma = 1$, i.e., $\pi(y|x) \propto \exp\left\{-\frac{(y-x)^2}{2}\right\}$. We choose as weight functions $\omega(x, y) = [p(x)]^\theta$, with $\theta = 1/2$. Note that they cannot be obtained using the analytic form necessary in the standard MTM (Liu et al. 2000). Moreover, we consider four possible combinations of the $\beta(x, y)$ and $\gamma(x, y)$ functions

$$\begin{aligned}
\alpha_{1,1}(x, y) &= \beta_1(x, y)\gamma_1(x, y),\\
\alpha_{1,2}(x, y) &= \beta_1(x, y)\gamma_2(x, y),\\
\alpha_{1,3}(x, y) &= \beta_1(x, y)\gamma_3(x, y),\\
\alpha_{2,3}(x, y) &= \beta_2(x, y)\gamma_3(x, y),
\end{aligned}$$

where each $\beta_i(x, y)$, $i = 1, 2$, and $\gamma_j(x, y)$, $j = 1, 2, 3$, are defined in Sects. 4.1 and 4.2. Then, we run the different MTM algorithms with $N = 10$ and $N = 100$ candidates. Table 7 shows the acceptance rate (the averaged probability of accepting a movement) and normalized linear correlation coefficient (between one state of the chain and the next) averaged over 2,000 runs and obtained with the different techniques where $N = 10$.

Table 8 illustrates the results using $N = 100$. We observe that $\alpha_{1,3}$ provides that greatest acceptance rate and lowest correlation in both cases. The acceptance rate of $\alpha_{1,1}$ decreases with $N = 100$ because of $\gamma_1(x, y|\mathbf{x}^*_{-k}, \mathbf{y}_{-k}) = W_x$ diminishes with the number of tries N. Moreover, the correlation appears (almost) invariant with the number of tries N.

Better performances can be attained using the acceptance function of (Pandolfi et al. 2010) and rewritten in Eq. (3), as expected analyzing the analytic form of the different acceptance functions. Indeed, we obtain acceptance rates of 0.74, 0.81 and correlation 0.96, 0.96 with $N = 10$ and $N = 100$, respectively.

6.6 Smiling-face distribution

In this section, we show that the power of the MTM schemes increases when they draw from more complicated target distributions in higher dimensions, w.r.t. a standard MH algorithm. To provide a graphical example, we consider a bidimensional target pdf $p_o(\mathbf{x})$ (where $\mathbf{x} = [x^{(1)}, x^{(2)}]^T \in \mathbb{R}^2$, $x^{(i)} \in \mathbb{R}$, $i = 1, 2$) composed as a mixture of 4 densities,

$$p_o(\mathbf{x}) \propto \frac{1}{4} \sum_{i=1}^{4} p_i(\mathbf{x}). \tag{20}$$

The first three components are proportional to bivariate Gaussian pdfs, i.e.,

$$p_i(\mathbf{x}) = p_i(x^{(1)}, x^{(2)}) = \exp\left\{ -\frac{\left(x^{(1)} - \mu_i^{(1)}\right)^2}{2\left(\sigma_i^{(1)}\right)^2} - \frac{\left(x^{(2)} - \mu_i^{(2)}\right)^2}{2\left(\sigma_i^{(2)}\right)^2} \right\},$$

with $i = 1, 2, 3$, $\mu_1^{(1)} = -7$, $\mu_1^{(2)} = 35$, $\mu_2^{(1)} = 7$, $\mu_2^{(2)} = 35$, $\mu_3^{(1)} = 0$, $\mu_3^{(2)} = 23$, $\sigma_1^{(1)} = 2$, $\sigma_1^{(2)} = 2$, $\sigma_2^{(1)} = 2$, $\sigma_2^{(2)} = 2$, $\sigma_3^{(1)} = 1$ and $\sigma_3^{(2)} = 4$. The last component is a banana-shaped density (Haario et al. 1999; Lan et al. 2012), i.e.,

$$p_4(\mathbf{x}) = p_4(x^{(1)}, x^{(2)}) = \exp\left\{ -\frac{\left(x^{(1)}\right)^2}{\eta} - \frac{\left(x^{(1)} - \rho\left(x^{(2)}\right)^2 + 100\rho\right)^2}{2} \right\},$$

with $\eta = 144.5$ and $\rho = 0.08$. The banana-shaped distribution was first introduced in Haario et al. (1999) and is known in literature to be a difficult target. This kind of bidimensional and multimodal mixtures of densities is often used to compare the performance of different MCMC techniques (Liang et al. 2010, Chapter 5; Haario et al. 1999, 2001; Lan et al. 2012). The parameters of the Gaussian components and the banana-shaped pdf are chosen in order to form a "smiling face" as illustrated in Fig. 4a. The reason is that, in this way, it is possible to illustrate *graphically* the performance of different samplers, as we show below.

To draw from $p_o(\mathbf{x})$, we apply a MH and a MTM scheme using for both a random walk Gaussian proposal pdf, i.e.,

$$\pi(\mathbf{x}_t|\mathbf{x}_{t-1}) \propto \exp\left\{ -\left(x_t^{(1)} - x_{t-1}^{(1)}\right)^2 \Big/ (2\sigma_p^2) - \left(x_t^{(2)} - x_{t-1}^{(2)}\right)^2 \Big/ (2\sigma_p^2) \right\}.$$

In order to show the speed of the convergence of the samplers, we have generated only 500 samples with a MTM with different number of candidates $N = 1, 5, 100, 1{,}000$ (note with $N = 1$ is a standard MH) and different standard deviation $\sigma_p = 5, 10$ of the proposal.

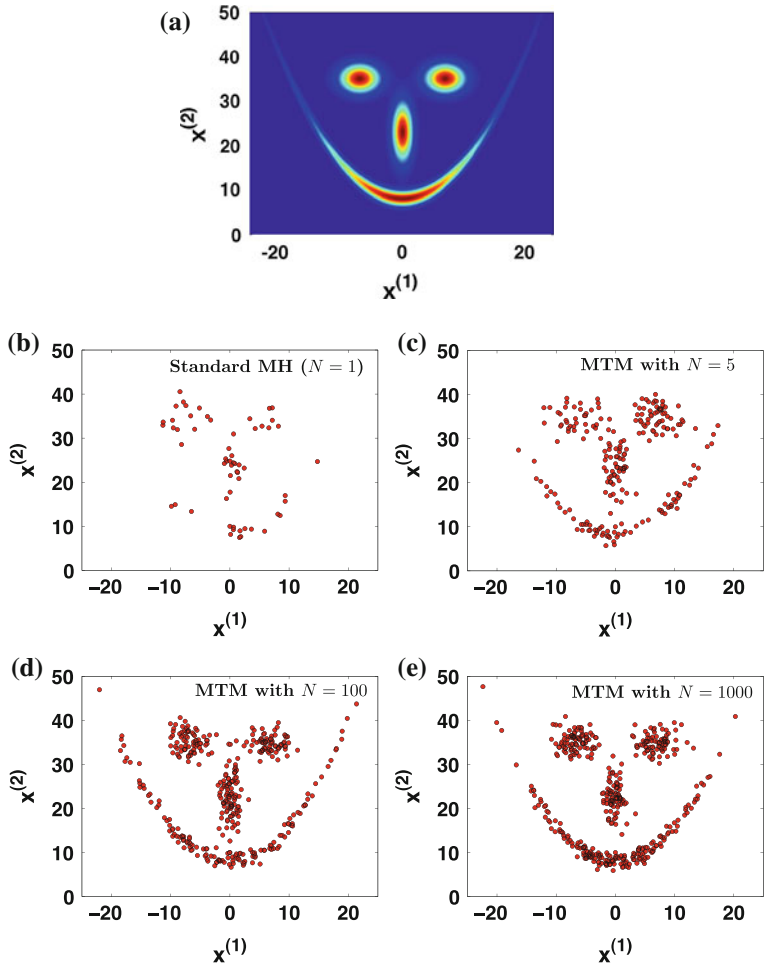

Fig. 4 **a** The Smiling-Face target density. The remaining figures **b**–**e** depict the first 500 generated samples drawn from the different samplers in one run (with $\sigma_p = 10$). Note that the number of points are less than 500 since, in certain iterations, the chain remains in the same state (depending on the acceptance probability α) so that some points are repeated. **b** Samples generated by a standard MH ($N = 1$). **c** Samples generated by a MTM with $N = 5$. **d** Samples generated by a MTM with $N = 100$. **e** Samples generated by a MTM with $N = 1{,}000$. It is evident that the MTM scheme speeds up the convergence of the Markov chain

Tables 9, 10 provide the average acceptance probability of a new state in the first column (the averaged values of α), the jump rate among different modes in the second column (from "left eye" to the "smile", or from the "smile" to the "nose" etc.) and the linear correlation for each component of $\mathbf{x}$, in the last column. To compute the mode-jump rate we establish that the state $\mathbf{x}_t$ belongs to the mode i^* if

$$i^* = \arg \max_{i \in \{1,\ldots,4\}} p_i(\mathbf{x}_t),$$

Table 9 Numerical results with $\sigma_p = 5$

Number of tries N	Acceptance rate	Mode-jump rate	Correlation
$N = 1$ (standard MH)	0.2296	0.0401	$x^{(1)} \to 0.9460$
			$x^{(2)} \to 0.9749$
$N = 5$	0.5118	0.1166	$x^{(1)} \to 0.8661$
			$x^{(2)} \to 0.9492$
$N = 100$	0.7137	0.3373	$x^{(1)} \to 0.6193$
			$x^{(2)} \to 0.8508$
$N = 1{,}000$	0.7919	0.4430	$x^{(1)} \to 0.4724$
			$x^{(2)} \to 0.7662$

Table 10 Numerical results with $\sigma_p = 10$

Number of tries N	Acceptance rate	Mode-jump rate	Correlation
$N = 1$ (standard MH)	0.1464	0.0598	$x^{(1)} \to 0.9097$
			$x^{(2)} \to 0.9653$
$N = 5$	0.4207	0.2313	$x^{(1)} \to 0.7536$
			$x^{(2)} \to 0.8454$
$N = 100$	0.7670	0.5020	$x^{(1)} \to 0.3570$
			$x^{(2)} \to 0.4607$
$N = 1{,}000$	0.8930	0.6520	$x^{(1)} \to 0.1635$
			$x^{(2)} \to 0.1453$

where $p_i(\mathbf{x}_t)$ are the 4 components in the mixture of Eq. (20). All results are averaged over 2000 runs using $\sigma_p = 5$ in Table 9 and $\sigma_p = 10$ in Table 10.

From the tables, we can observe that the MTM clearly outperforms the standard MH since, as N grows, the correlation decreases and the mode-jump rate increases (as does the acceptance rate) regardless of the chosen parameter σ_p of the proposal. Obviously, the mode-jump rate is always less than the average value of the probability α of accepting a movement (the acceptance rate), since the mode-jumps represent a subset of all accepted movements. Moreover, the standard deviation $\sigma_p = 10$ of the proposal pdf works better for the MTM method. In general, the MTM schemes work better with huge scaling parameters and a great-enough number of candidates N (see also the discussion in the next section).

Figures 4b–e depict generated samples over one run. Clearly, in general we observe less than 500 points since in certain cases a new movement is rejected and the chain remains in the same state. Namely, certain points are repeated. This effect is evident with the standard MH ($N = 1$) whereas it vanishes as the number of candidates N grows. Moreover, with greater N, the number of jumps among different modes also increases quickly. As a consequence, with the MTM technique ($N = 5, 100,$

1,000) all the features of the "face" (our target pdf) are completely described since the convergence of the chain is clearly speeded up. Therefore, with this numerical example, the main advantage of a MTM method becomes apparent: it can explore a larger portion of the sample space without a decrease of the acceptance rate, or even an increase thereof.

7 Discussion

In this work, we have studied the flexibility in the design of MTM techniques. We have introduced a MTM with generic weight functions (the analytic form can be chosen arbitrarily) and different proposal densities (each candidate can be drawn from a different pdf) combining the algorithms in Casarin et al. (2013) and Pandolfi et al. (2010). Moreover, we have proposed a general framework for construction of acceptance probabilities in the MTM schemes, providing also specific examples. Finally, we have also designed a MTM algorithm without the need to randomly generate the reference points (Robert 2012). We have proved that the novel techniques satisfy the detailed balance condition, and carried out numerical simulations. Observing the theoretical workings and the numerical results, we can make the following conclusions and observations:

1. *General considerations:* The classical MTM method, proposed in (Liu et al. 2000), clearly outperforms the standard MH algorithm using the same proposal pdf, in the sense that as the number of candidates increases, $N \to \infty$, then the correlation decreases quickly to zero (see Sect. 6.3 for further considerations). If a designed MTM scheme does not fulfill this property, then it is totally useless since the computational cost increased but the performance is not improved. Suitable MTM methods can be applied efficiently to any kind of target distributions (bounded or unbounded, with heavy tails or not), as shown in our numerical simulations (see Sect. 6.4). Moreover, the advantages of using a MTM technique w.r.t. a standard MH algorithm clearly grow as the dimensionality of the target increases.
2. *MTM schemes as black-box algorithms:* the numerical simulations show that, with a suitable number of tries N, the MTM methods provide good results independently of the choice of the parameters of the proposal. Therefore, it is important to remark that, even if no information about the target is available (for instance, about the location of the modes), a MTM scheme allows the use of a proposal pdf with a huge scaling parameter in order to explore quickly different regions of the space. Indeed, using a great-enough number of tries, this black-box approach is quite robust and always gives satisfactory performance. On other hand, with a huge scaling parameter, a standard MH usually produces a very small rate of jumps and, as a consequence, a very high correlation.
3. *Choice of the weights:* the possibility to choose any bounded and positive weight functions makes the MTM scheme easier to design since the user does not need to check any conditions of the weights (for instance symmetry of the function λ) independently of the choice of proposal pdf. Namely, the proposal distribution and the weight functions can be selected separately, to fit well to the specific problem and to improve the performance of the technique. Note that, in some

MTM approaches the symmetry condition of the function λ can be complicated, see for instance (Martino et al. 2012a; Qin and Liu 2001).

Further theoretical or numerical studies are needed to determine the best choice of weight functions given a certain proposal and target density. We find that the weights of the analytic form proposed in Liu et al. (2000) (see for instance Eq. 4) usually provide better results. Within this class, the importance weights $\omega_i(y_i) = \frac{p(y_i)}{\pi_i(y_i|x)}$, based on the importance sampling principle (Liu 2004; Robert and Casella 2004), appear to be a good choice in theory. Numerical results also suggest that weights simply proportional to the target density $\omega_i(y_i) = p(y_i)$ can provide good performance. In (Bédard et al. 2012) the authors note that importance weights place higher probability on selecting candidates that are further away from the current state of the chain, but finally they prefer to use weights proportional to the target density based on numerical results.

If the evaluation of the target $p(x)$ is computationally expensive such that the target function can not be included in the calculations of the weights, then the weight functions of the analytic class $\omega_i(y_i, x) = p(y_1)\pi_i(x|y_i)\lambda(x, y_i)$ proposed in Liu et al. (2000) cannot be used. Indeed, it is impossible to find a symmetric function $\lambda(x, y) = \lambda(y, x)$ in order to remove the dependence on $p(x)$ in the weights (in this case there is just one possibility that $p(x)$ is constant, i.e., $p(x) = p(y)$ for all $x, y \in \mathcal{D}$). In this case, a possible choice of the weights can be proportional to the proposal pdfs, namely $w(y_i) = \pi(x|y_i)$ for instance. Clearly, it is not the optimal choice but, also in this case, the MTM can easily help to explore a larger portion of the sample space w.r.t. standard MH (see Sect. 6.2).

4. *Use of different proposal pdfs:* a MTM scheme with different proposal densities can be a very powerful framework mainly to tackle applications with high dimensionality and target distributions with several modes. In our opinion, the most promising scenario is to use different independent proposal distributions updating certain parameters (as mean and variance) each iteration of the chain, or selecting the best proposal among a set of functions (see Sect. 6.3 for further considerations). In this adaptive framework, the independent proposal pdfs could improved to fit better w.r.t. the target. This scheme has not been already exploited completely. It is important to remark that, in order to obtain a fair comparison among the generated candidates, it is recommendable to use proposal functions with the same area below, for instance they can be normalized.
5. *Flexibility of the acceptance probabilities:* we have shown that there is certain freedom in the design of a MTM algorithm, specifically in the choice of the acceptance probability α. This is also confirmed by other works in literature that design suitable MTM schemes with correlated candidates but they are quite different (the strategies in Martino et al. 2012a; Qin and Liu 2001 generate the candidates sequentially, whereas the approach in Craiu and Lemieux 2007 uses a block philosophy). However, although the detailed balance condition is always satisfied in all cases, the performance is different. Numerical results suggest that α functions as close as possible to the standard MTM method (Liu et al. 2000), using also the weights of the analytic form in Eq. (4), provide better results. Similar considerations can be made with respect to the standard MH algorithm (Barker 1965; Hastings 1970; Peskun 1973).

6. *Reference points:* we have described a possible MTM algorithm without drawing reference points. As seen in the numerical results, in this case it seems there is an optimal number of candidates N. As $N \to \infty$ the performance becomes very poor. Therefore, we can figure out that the "secret" of the good performance of the standard MTM scheme in (Frenkel and Smit 1996; Liu et al. 2000) is contained in the random generation of the reference points. However, there exists an important special case where the reference points are completely unnecessary: using independent proposal densities. In this case, the reference points can be set deterministically, equal to the previous generated candidates. This scheme, using just one proposal (drawing N candidates from the same pdf) jointly with importance weights, appears as the easiest and natural procedure to combine the classical MH algorithm and importance sampling (Robert and Casella 2004) (see Fig. 3a).
7. *Number of candidates:* All the schemes proposed in literature and also in this work use a fixed number of candidates N. An important improvement would consist on tuning adaptively the number N depending on the discrepancy between target and proposal distributions. To do this, a certain measure is needed, for instance, as the *effective sample size* of the importance sampling framework (Liu 2004; Robert and Casella 2004). Clearly, this idea could be more effective using independent proposal pdf since it is necessary to measure the discrepancy between the proposal and the target functions (with a random walk, for instance, the mean of the proposal changes each step and the distance w.r.t. the target varies as well). Another possibility could be to combine MTM and the delayed rejection method (Mira 2001; Tierney and Mira 1999). With this kind of procedures, the optimal trade off between computational cost and performance would be achieved.

Acknowledgments We would like to thank the Reviewers for their comments which have helped us to improve the first version of manuscript. Moreover, this work has been partially supported by Ministerio de Ciencia e Innovacin and by the Ministerio de Economía of Spain (project MONIN, ref. TEC-2006-13514-C02- 01/TCM, project COMONSENS, id. CSD2008-00010, project DEIPRO ref. TEC2009-14504-C02-01, project ALCIT, id. TEC2012-38800-C03-01 and project COMPREHENSION, id. TEC2012-38883-C02-01) and Comunidad Autonoma de Madrid (project PROMULTIDIS-CM, ref. S-0505/TIC/0233).

References

Barker AA (1965) Monte Carlo calculations of the radial distribution functions for a proton–electron plasma. Aust J Phys 18:119–133

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

Brooks SP (1998) Markov Chain Monte Carlo method and its application. J R Stat Soc Ser D (The Statistician) 47(1):69–100

Casarin R, Craiu R, Leisen F (2013) Interacting multiple try algorithms with different proposal distributions. Stat Comput 23(2):185–200. doi:10.1007/s11222-011-9301-9

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

Devroye L (1986) Non-uniform random variate generation. Springer, Berlin

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
Gamerman D, Lopes HF (2006) Markov Chain Monte Carlo: stochastic simulation for Bayesian inference. Chapman and Hall/CRC, Boca Raton
Gilks WR, Richardson S, Spiegelhalter D (1995) Markov Chain Monte Carlo in practice: interdisciplinary statistics. Taylor & Francis, Inc., UK
Haario H, Saksman E, Tamminen J (1999) Adaptive proposal distribution for random walk Metropolis algorithm. Comput Stat 14:375–395
Haario H, Saksman E, Tamminen J (2001) An adaptive Metropolis algorithm. Bernoulli 7(2):223–242
Hastings WK (1970) Monte Carlo sampling methods using Markov chains and their applications. Biometrika 57(1):97–109
Lan S, Stathopoulosy V, Shahbaba B, Girolami M (2012) Langrangian dynamical Monte Carlo. arXiv:1211.3759v1
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 (2012a) A multi-point Metropolis scheme with generic weight functions. Stat Probab Lett 82(7):1445–1453
Martino L, Read J, Luengo D (2012b) Improved adaptive rejection Metropolis sampling algorithms. arXiv:1205.5494v4
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
Mira A (2001) On Metropolis–Hastings algorithms with delayed rejection. Metron 59:231–241
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
Peskun PH (1973) Optimum Monte-Carlo sampling using Markov Chains. Biometrika 60(3):607–612
Qin ZS, Liu JS (2001) Multi-point Metropolis method with application to hybrid Monte Carlo. J Comput Phys 172:827–840
Robert CP (2012) "Xi' An's Og, an attempt at bloggin... " Blog (by Christian P. Robert). http://xianblog.wordpress.com/2012/01/23/multiple-trypoint-metropolis-algorithm/
Robert CP, Casella G (2004) Monte Carlo statistical methods. Springer, Berlin
Roberts GO, Gelman A, Gilks WR (1997) Weak convergence and optimal scaling of random walk Metropolis algorithms. Ann Appl Probab 7:110–120
Storvik G (2011) On the flexibility of Metropolis–Hastings acceptance probabilities in auxiliary variable proposal generation. Scand J Stat 38(2):342–358
Tierney L (1994) Markov chains for exploring posterior distributions. Ann Stat 33:1701–1728
Tierney L, Mira A (1999) Some adaptive Monte Carlo methods for Bayesian inference. Stat Med 18:2507–2515
Zhang Y, Zhang W (2012) Improved generic acceptance function for multi-point Metropolis algorithm. In: 2nd International conference on electronic and mechanical engineering and information technology (EMEIT-2012)