# Parallel Tempering with Equi-Energy Moves

Meili Baragatti, Agnès Grimaud, Denys Pommeret

# Parallel Tempering with Equi-Energy Moves

Meïli Baragatti[1,2,*], Agnès Grimaud[2], Denys Pommeret[2]

[1] *Ipsogen SA, Luminy Biotech Entreprises, Case 923, Campus de Luminy, 13288 Marseille Cedex 9, France.*

[2] *Institut de Mathématiques de Luminy (IML), CNRS Marseille, case 907, Campus de Luminy, 13288 Marseille Cedex 9, France.*

[*] *baragatt@iml.univ-mrs.fr, baragattimeili@hotmail.com.*

PREPRINT 25-01-2011

**Abstract**

The Equi-Energy Sampler (EES) introduced by Kou et al. [2006] is based on a population of chains which are updated by local moves and equi-energy jumps. This algorithm has been developed to facilitate global moves between the different chains, resulting in a good exploration of the states space by the target chain. This method seems to be more efficient than the classical Parallel Tempering (PT) algorithm. However it necessitates increased storage and the convergence of the original EES is not guaranteed (see Andrieu et al. [2008])). In this paper we propose an adaptation of the EES that combines PT with the principle of jumping between chains with same levels of energy. This adaptation, that we shall call Parallel Tempering with Equi-Energy Moves (PTEEM), keeps the original idea of the EES method and ensures convergence. Performances of the PTEEM algorithm are compared with those of the EES and of the standard PT algorithm in the context of mixture models.

*Keywords*: Algorithm convergence, equi-energy sampler, mixture models, parallel tempering, population-based MCMC.

## 1 Introduction

A common problem in Bayesian statistics is that of generating random variables from a target density $\pi$. Many solutions have been proposed in the last two decades, deriving essentially from the Monte Carlo Markov Chains (MCMC) approach introduced by Metropolis et al. [1953] and Hastings [1970]. In classical MCMC methods, a Markov process is built to sample the target probability distribution. But in practice, the Markov process can be easily trapped into a local maximum from where it cannot escape in reasonable time (see for instance Liang and Wong [2001]). Many techniques have been proposed to address this waiting time problem, including among others Parallel Tempering (PT) (see Geyer and Thompson [1995]), and more recently Equi-Energy Sampler (EES) (Kou et al. [2006]). In this paper we focus on these two methods and we propose an adaptation that can be seen as a combined version of PT and EES algorithms, and that we called the Parallel Tempering with Equi-Energy Moves (PTEEM) algorithm. Before developing this method, PT and EES algorithms are briefly recalled.

On some state space $\mathcal{X}$ with associated $\sigma$-algebra $\mathcal{B}(\mathcal{X})$, the target density is proportional to

$$\pi(x) \quad \propto \quad \exp\{-h(x)\},$$

where $h(x)$ denotes the energy function. In a classical Metropolis-Hasting algorithm a new state $y$ is generated from a current state $x$ of the Markov process by drawing $y$ from a proposal transition function $q(x; y)$. The new state $y$ is accepted with the probability $\min(1, r)$, where $r$ is the Metropolis-Hastings ratio:

$$r \quad = \quad \frac{\pi(y)q(y;x)}{\pi(x)q(x;y)}.$$

The Markov process converges to the target distribution $\pi$ using any positive transition function $q(x; y)$ and starting from any initial configuration. Nevertheless, in practice, the Markov process can be trapped into a deep local minimum of energy. To avoid this problem the principle of PT is to choose $N$ temperatures $T_1 = 1 < T_2 < \cdots < T_N$, and to run in parallel $N$ associated MCMC chains having different stationary distributions, $\pi_1, \cdots, \pi_N$, where

$$\pi_i \quad \propto \quad \pi^{1/T_i}.$$

Since the tempered distribution becomes flatter as the temperature increases, the chains at high temperatures can move more freely between modes. The PT algorithm consists of two steps at each iteration: a parallel step updating every chain by using their respective MCMC algorithm, and a swapping step consisting in choosing randomly two chains and in proposing a swap between those. The probability of accepting the swap between two chains, say $i$ and $j$, is

$$\min\left\{1, \frac{\pi_i(x_j)\pi_j(x_i)}{\pi_i(x_i)\pi_j(x_j)}\right\},$$

where $x_i$ stands for the state of the $i$th chain. An advantage of the PT algorithm is its ability to use information from different chains through the swapping step. Therefore the swapping step allows the chain associated with the posterior distribution to escape from its local modes, thereby improving mixing. Some improvements of PT have been proposed as the delayed rejection (see Green and Mira [2001]) which permits to propose a new chain candidate when the first swap did not occur. However PT algorithm does not retain information of where chains have been and it does not take into account all chains to choose one of the best swap. This is what is done by the EES proposed by Kou et al. [2006]. Note that the method of Atchadé and Liu [2006], called multicanonical sampling, is in the same spirit.

In EES a sequence of $K + 1$ energy levels and a sequence of $K$ temperatures are introduced:

$$H_1 < H_2 < \cdots < H_{K+1} = \infty, \quad \text{and} \quad T_1 = 1 < T_2 < \cdots < T_K,$$

such that $H_1 \leq \min(h(x))$. The EES considers a population of $K$ distributions, each indexed by a temperature and an energy truncation. The probability density function of the $i$th distribution is

$$\tilde{\pi}_i(x) \quad \propto \quad \exp\{-h_i(x)\},$$

where

$$h_i(x) \quad = \quad \frac{\max(h(x), H_i)}{T_i}.$$

The method begins by sampling the $K$th chain from a Metropolis-Hastings kernel with stationary distribution $\tilde{\pi}_K$. Once convergence is reached, samples are stored and the next chain targeting $\tilde{\pi}_{K-1}$ starts. At each step all chains are updated by either (with a fixed probability $p_{ee}$) using a Metropolis-Hastings kernel or (with probability $1-p_{ee}$) by proposing to exchange the current state of the chain with a value from the past of the previous chain. This exchange between chains is called the equi-energy jump: two successive chains are considered, say chains $i$ and $i-1$, and a state $y$ is chosen from the chain $i$ such that $h(y)$ and $h(x_{i-1})$ belong to the same energy ring $D_k = [H_k, H_{k+1})$ for some $k \in \{1, \cdots, K\}$. Then $y$ is accepted to be the next state of the $(i-1)$th chain with probability

$$\min\left\{1, \frac{\tilde{\pi}_{i-1}(y)\tilde{\pi}_i(x_{i-1})}{\tilde{\pi}_{i-1}(x_{i-1})\tilde{\pi}_i(y)}\right\},$$

The EES continues the construction of the others chains in much the same way, until it targets $\tilde{\pi}_1 = \pi$, which is the target density of interest.

The advantage of the EES is that it retains information of all chains and it is able to make large moves between separated modes within energy rings. Moreover, it seems to be very efficient compared to classical MCMC methods as PT (see Kou et al. [2006]). But a possible weakness of the EES is the cost of increased storage, all the past being taken into account in energy rings. In addition the proof of the convergence appears incomplete (see Andrieu et al. [2007, 2008]), and Atchadé et al. [2010] showed that the asymptotic variance of the EES can be substantially different than that suggested by Kou et al. [2006]. Note that recently Hua and Kou [2010] completed the proof of the convergence of the EES in the case of a countable state space.

We propose to address some drawbacks of EES by adapting the concept of equi-energy jump in a schema of PT, with rings of energy depending only on the currents chains, not on all the past. Then the jump between chains still depends on their energies, and the candidate chains for swapping are chosen randomly and uniformly. This combination of EES and PT yields to the new algorithm PTEEM for which convergence is ensured (see Propositions 2.1 and 2.2). The possible loss or gain of this algorithm compared to EES and PT are evaluated through simulations and real data.

The paper is organized as follows: In Section 2 the new PTEEM algorithm and some theoretical properties are presented. In Sections 3 and 4 comparisons between PTEEM, EES and PT are presented in the case of Metropolis-Hastings and Gibbs sampler algorithms. Two cases of mixture models are studied through simulations and real data.

# 2 PTEEM algorithm

## 2.1 Description of the algorithm

We fix $d+1$ energy levels $H_1 < H_2 < \cdots < H_{d+1} = \infty$ and $N$ temperatures $T_1 < T_2 < \cdots < T_N$, with $T_1 = 1$ and $H_1 \leq \min(h(x))$. The algorithm considers a population of $N$ chains associated with distributions $\pi_i(x) \propto \pi(x)^{1/T_i}$, each $\pi_i$ being defined on some state space $\mathcal{X}$ with associated $\sigma$-algebra $\mathcal{B}(\mathcal{X})$. Clearly $\pi_1 = \pi$.

**Remark 2.1** *Kou et al. [2006] used $\tilde{\pi}$, a truncation of the energy function, which leads to intractable simulations in the case of Gibbs sampling. In PTEEM we use $\pi$ and not $\tilde{\pi}$, allowing us to apply the algorithm within a Gibbs sampler framework (see Section 4).*

The energy rings are constructed as in Kou et al. [2006], except for the first one. The state space $\mathcal{X}$ is partitioned according to the energy levels: $\mathcal{X} = \bigcup_{j=1}^{d} D_j$, where

$$\begin{aligned} D_j &= \{x \in \mathcal{X}; h(x) \in [H_j, H_{j+1})\}, \quad j = 2, \cdots, d \\ D_1 &= \{x \in \mathcal{X}; h(x) \in (-\infty, H_2)\}. \end{aligned}$$

Each step of the PTEEM algorithm is decomposed into two types of moves: local moves via a classical MCMC algorithm and global moves allowing an exchange between two chains with similar energy.

**Local moves** At each step a new state $y_i$ is proposed to the $i$th chain, for all $i = 1, \cdots, N$, using MCMC algorithm. When using a Gibbs sampler, the $i$th chain takes the value $y_i$. When using Metropolis-Hasting algorithm, the current value $x_i$ of the $i$th chain is replaced by $y_i$ with probability

$$\alpha = \min\left(1, \frac{f(y_i)q(y_i; x_i)}{f(x_i)q(x_i; y_i)}\right),$$

with $\pi(x) = f(x)/K$, where the normalizing constant $K$ may not be known. Assuming that the proposal distribution is symmetric, i.e., $q(x; y) = q(y; x)$, the local move is accepted with probability $\alpha = \min(1, \frac{f(y_i)}{f(x_i)})$.

**Global moves** At each step, an energy ring $D_j$ containing at least two chains is chosen randomly. Two chains are then chosen uniformly in $D_j$, say the $i$th and the $k$th ones (with $i < k$), and an exchange move is proposed between the actual two states of these chains. The move is from $s = (x_1, \cdots, x_i, \cdots, x_k, \cdots, x_N)$ to $s' = (x_1, \cdots, x_k, \cdots, x_i, \cdots, x_N)$.

Writing $\pi^*(s) = \prod_{i=1}^{N} \pi_i(x_i)$ on $(\mathcal{X}^N, \mathcal{B}(\mathcal{X}^{\mathcal{N}})$, the probability acceptance for the global move is given by:

$$\begin{aligned} \rho(s; s') &= \min\left\{1, \frac{\pi^*(s')}{\pi^*(s)}\right\} \\ &= \min\left\{1, \frac{\pi_i(x_k)\pi_k(x_i)}{\pi_i(x_i)\pi_k(x_k)}\right\}. \end{aligned} \tag{1}$$

The chains are not Markov by themselves, it is the whole stochastic process made of the $N$ chains together that forms a Markov chain on $(\mathcal{X}^N, \mathcal{B}(\mathcal{X}^{\mathcal{N}}))$.

**Remark 2.2** *It is of interest to compare the total number of local and global moves required in PTEEM and EES algorithms. Let us denote by $B$ the size of the burn-in period, by $R$ the number of iterations necessary to initialize energy rings within EES, and by $M$ the sample size of the chains (after the burn-in period). We have:*

- *For EES, the total number of local moves is equal to*

$$K(B+R)+M+(1-p_{ee})\left(\frac{(K-1)K}{2}(B+R)+(K-1)M\right),$$

*and the total number of global moves is equal to*

$$p_{ee}\left(\frac{(K-1)K}{2}(B+R)+(K-1)M\right),$$

*where $K$ denotes the number of chains in EES.*

- *For PTEEM, the total number of local moves is $NM+NB$,*
*and the total number of global moves is $M+B$,*
*where $N$ stands for the number of chains in PTEEM.*

*Then in terms of simulations number, PTEEM necessitates more moves than EES when $M$ is large since usually $N$ is larger than $K$. In terms of storage, to obtain the $(i+1)$th iteration of the target chain, EES uses $K(B+R)+i+(1-p_{ee})\left(\frac{(K-1)K}{2}(B+R)+(K-1)i\right)$, values in memory to choose an element in an energy ring, whereas PTEEM necessitates only $N$ values. From our experience, CPU time to compute one iteration increases within EES as the simulations go along. In opposition CPU time for one iteration is constant within PTEEM algorithm.*

## 2.2 Some theoretical results

Denote by $S$ the Markov chain on $(\mathcal{X}^N, \mathcal{B}(\mathcal{X}^{\mathcal{N}}))$ obtained by the PTEEM algorithm, a state of $S$ is written $s$. The transition kernel associated with an iteration of PTEEM is written $P$, and $P^k$ is the $k$-step transition kernel. They are defined on $\mathcal{X}^N \times \mathcal{B}(\mathcal{X}^{\mathcal{N}})$. The transition kernel associated with the local move of the $i$th chain is written $PL_i$, and is defined on $\mathcal{X} \times \mathcal{B}(\mathcal{X})$. The transition kernel associated with the $N$ local moves of an iteration of PTEEM is written $PL$, and is defined on $\mathcal{X}^N \times \mathcal{B}(\mathcal{X}^{\mathcal{N}})$. The transition kernel associated with the equi-energy move is written $PE$, and is defined on $\mathcal{X}^N \times \mathcal{B}(\mathcal{X}^{\mathcal{N}})$. Writing

$$\begin{aligned} s &= (x_1, \ldots, x_i, \ldots, x_k, \ldots, x_N) \\ s' &= (x'_1, \ldots, x'_i, \ldots, x'_k, \ldots, x'_N), \end{aligned}$$

we have

$$\begin{aligned} PL(s,s') &= \prod_{i=1}^{N} PL_i(x_i, x'_i) \\ P(s,s') &= (PE * PL)(s,s') = \int_{\mathcal{X}^N} PE(\tilde{s}, s') PL(s, \tilde{s}) d\tilde{s} \end{aligned}$$

Write $q(s,s')$ the auxiliary distribution to propose $s'$ from $s$ in an equi-energy move, and $q_i(x_i, x'_i)$ the auxiliary distribution to propose $x'_i$ from $x_i$ in a local move of the $i$th chain. The total variation norm for a measure $\mu$ on $(\mathcal{X}^N, \mathcal{B}(\mathcal{X}^{\mathcal{N}})$ is defined by:

$$\|\mu\|_{TV} = \sup_{A \in \mathcal{B}(\mathcal{X}^{\mathcal{N}})} |\mu(A)|.$$

**Proposition 2.1** *If the transition kernels associated with the local moves are $\pi_i$-irreducible and aperiodic with stationary distributions $\pi_i$, $i = 1, \cdots, N$, then we have for $\pi^*$-almost all $s \in \mathcal{X}^N$*

$$\lim_{n\to\infty} \|P^n(s,.) - \pi^*\|_{TV} \quad = \quad 0.$$

*Therefore $\pi^*$ is the stationary distribution of $S$ and the chain associated with $T_1 = 1$ provides samples corresponding to $\pi_1 = \pi$, which is the target distribution.*

**Proof**: in appendix B.1.

**Remark 2.3** *In Proposition 2.1, the transition kernels of the local moves are assumed to be aperiodic. We can relax this hypothesis. In fact, it is sufficient that only one of the $N$ transition kernel is aperiodic to have $P$ aperiodic.*
*However, all transition kernels should be irreducible to have $P$ irreducible.*

This proposition has minimal assumptions, which are usually not hard to verify. However, it is possible to have a null set of states from which convergence does not occur. The following lemma and proposition have stronger assumptions that ensure convergence from all starting points.

**Lemma 2.1** *Assume that the transition kernels associated with the local moves are aperiodic with stationary distributions $\pi_i$ $(i = 1, \ldots, N)$, and that $\forall (x_i, x_i') \in \mathcal{X} \times \mathcal{X}$ we have $q_i(x_i, x_i') > 0$, $i \in \{1, \ldots, N\}$. Then the chain $S$ is $\pi^*$-irreducible, positive and Harris-recurrent.*

**Proof**: in appendix B.2.

**Proposition 2.2** *Assume that the transition kernels associated with the local moves are aperiodic with stationary distributions $\pi_i$ $(i = 1, \ldots, N)$, and that $\forall (x_i, x_i') \in \mathcal{X} \times \mathcal{X}$ we have $q_i(x_i, x_i') > 0$, $i \in \{1, \ldots, N\}$. Then we have for all $s \in \mathcal{X}^N$*

$$\lim_{n\to\infty} \|P^n(s,.) - \pi^*\|_{TV} \quad = \quad 0.$$

**Proof**:
Using Lemma 2.1 and Proposition 2.1, $S$ is a Markov chain $\pi^*$-irreducible, aperiodic, with stationnary distribution $\pi^*$ and Harris-recurrent. The result follows from Theorem 1 of Tierney [1994].

The assumption $\forall (x_i, x_i') \in \mathcal{X} \times \mathcal{X}$, $q_i(x_i, x_i') > 0$ is quite strong. However, in our applications it is usually not hard to verify it.

### 2.3 Choice of energy ladder and temperatures

Following our experience we propose some choices of energy ladder and temperatures.

**Energy ladder** The levels $H_1, H_2, \ldots, H_d$ are associated with $d$ energy rings, the first one including states having an energy value lower than $H_2$ and including only few states having an energy value lower than $H_1$, and the last one including states having an energy value higher than $H_d$. Once the values $H_1$ and $H_d$ are chosen, the other energy levels can be set to be evenly spaced on a logarithmic scale

$$ln(H_i) = ln(H_1) + i\frac{ln(H_d) - ln(H_1)}{d-1}.$$

To choose $H_1$ and $H_d$ we use one or few runs of a classical MCMC algorithm with target density $\pi$. We take for $H_d$ the energy associated with a state with high enough finite energy compared to other states. Concerning $H_1$, we take the energy corresponding to an observed mode. In practice, we can take for $H_d$ the energy associated with a state after few iterations of the algorithm, and for $H_1$ the energy associated with a state after a burn-in period.

**Remark 2.4** *Concerning $H_1$, if the modes of the distribution of interest are known, we just have to take $H_1$ slightly lower than the energy of the highest mode.*

**Temperatures** The distribution associated with the highest temperature should be sufficiently flattened so that the associated chain can move freely from one mode to another. After choosing a $T_N$ value we just have to check that the associated chain moves easily. $T_1$ is obviously equal to 1, and is associated with the chain of interest. Once $T_1$ and $T_N$ are fixed, the other temperatures can be chosen by evenly spacing them on a logarithmic scale, or by evenly spacing the inverse temperatures geometrically (see for instance Kou et al. [2006] or Neal [1996]).

**Checking that the choices of temperatures and energy ladder are relevant** It is necessary to check on a run of PTEEM that the choices of temperatures and energy ladder are relevant. The chain 1 should have almost all its states in the first energy ring, the last chain should have almost all its states in the last energy ring, and between them the states of the different chains should be well distributed in the rings. The distribution in the rings can be considered as correct if there is no "energy gap" between adjacent chains, and if for each chain equi-energy moves are performed with several other chains. If poor mixing is observed between chains then it is necessary to adjust the temperatures or the energy levels, adding new temperatures for instance or proposing a new calibration. We illustrate such a problem in Table 1 and in an example of Section 4.1.

# 3 Comparisons in the case of local Metropolis-Hastings moves

To compare the three algorithms (PT, EES and PTEEM) when the local move is a Metropolis-Hastings algorithm, we consider sampling from a two-dimensional normal mixture model taken from Liang and Wong [2001] and used as an illustration by Kou et al. [2006]. Let

$$f(x) = \sum_{i=1}^{20} \frac{w_i}{\sigma_i\sqrt{2\pi}} exp\Big( -\frac{1}{2\sigma_i^2}(x-\mu_i)'(x-\mu_i)\Big),$$

where $\sigma_1 = \ldots = \sigma_{20} = 0.1$, $w_1 = \ldots = w_{20} = 0.05$, and the 20 mean vectors

$$(\mu_1, \ldots, \mu_{20}) = \begin{pmatrix} 2.18 & 8.67 & 4.24 & 8.41 & 3.93 & 3.25 & 1.70 & 4.59 & 6.91 & 6.87 \\ 5.76 & 9.59 & 8.48 & 1.68 & 8.82 & 3.47 & 0.50 & 5.60 & 5.81 & 5.40 \\ 5.41 & 2.70 & 4.98 & 1.14 & 8.33 & 4.93 & 1.83 & 2.26 & 5.54 & 1.69 \\ 2.65 & 7.88 & 3.70 & 2.39 & 9.50 & 1.50 & 0.09 & 0.31 & 6.86 & 8.11 \end{pmatrix}.$$

The different local modes are quite far from each other (most of them are more than 15 standard deviations from the nearest ones), hence this mixture distribution is quite challenging for sampling. In addition, the initial states of the different chains were drawn from a uniform distribution on $[0,1]^2$, a region far from the local modes.
Each algorithm was run 100 times. For each run, the PT and PTEEM algorithms were run for 2500 iterations after a burn-in period of 2500 iterations. Similarly, for each chain of the EES the burn-in period was of 2500 iterations, and for the first chain (the target chain) 2500 iterations were simulated after this burn-in period and the period to construct the rings, which was of 500 iterations. As in Kou et al. [2006], the Metropolis-Hastings proposal was a bivariate Gaussian $X_{n+1}^{(i)} \sim \mathcal{N}_2(X_n^{(i)}, \tau_i^2 I_2)$, with $\tau_i = 0.25\sqrt{T_i}$. Unlike them, the step size $\tau_i$ was not tuned later in the algorithms such that the acceptance ratio is in the range (0.22,0.32). Indeed, we would like to compare algorithms as simple as possible.
For the EES, we took the same number of chains, the same energy levels, the same temperatures and the same equi-energy jump probability than Kou et al. [2006] ($K = 5$, $H = (0.2, 2, 6.3, 20, 63.2)$, $T = (1, 2.8, 7.7, 21.6, 60)$, $p_{ee} = 0.1$). For the PT and PTEEM algorithms, $N = 20$ chains were taken, with temperatures between 1 and 60 evenly spaced on a logarithmic scale. As in Kou et al. [2006], the PT algorithm used a swap between neighboring temperature chains for the exchange operation, but only one swap was proposed at each iteration, to make it comparable with the PTEEM. For the PTEEM, the same 5 groups of energy than for the EES were taken.

The mean acceptances rates for the local Metropolis-Hastings moves and for the exchange moves between chains for the three algorithms are given in Table 2. In comparison Kou et al. [2006] obtained results slightly different probably because the step size $\tau_i$ was tuned in their EES.

To compare the ability of each algorithm to explore the distribution space, we considered for each run of each algorithm the number and frequency of visited modes by the target chain, as well as the estimations of the mean vector $(E(X_1), E(X_2))$ and of the second moments $(E(X_1^2), E(X_2^2))$ using the samples generated from the target chain. Table 3 contains these estimations. Concerning the estimations of the mean vector and of the second moments, the EES and PTEEM estimates were more accurate than those of the PT, with smaller mean squared errors. Moreover, it appeared that the PTEEM estimates were slightly more accurate than those of the EES. Concerning the number of visited modes, good results were obtained by the EES and PTEEM algorithms compared to the PT. The results are reported in Table 4. The mean number of visited modes by the PT on the 100 runs was 14.31, compared to 19.92 for the EES and 19.98 for the PTEEM. Then, as in Kou et al. [2006], we counted in each of the 100 runs for the three algorithms how many times the target chain visited each mode in the last 2500 iterations. The

absolute frequency error is given by $err_i = |\hat{f}_i - 0.05|$, where $\hat{f}_i$ is the sample frequency of the $i$th mode being visited ($i = 1, \ldots, 20$). The median and the maximum of $err_i$ over the 100 runs was calculated. To compare the three algorithms the ratios of these values between PT and EES, between PT and PTEEM and between EES and PTEEM were calculated for each mode. All these ratios are presented in Table 5. As denoted in Kou et al. [2006], EES seemed to be more efficient than PT (the mean of the ratios $R_{med(PT/EES)}$ over the 20 modes was 2.42, and the mean of the ratios $R_{max(PT/EES)}$ over the 20 modes was 2.92). As expected, PTEEM gave better results than PT (the mean of the ratios $R_{med(PT/PTEEM)}$ over the 20 modes was 2.52, and the mean of the ratios $R_{max(PT/PTEEM)}$ over the 20 modes was 3.07). Besides, we noticed a slight improvement of PTEEM compared to EES (the mean of the ratios $R_{med(EES/PTEEM)}$ over the 20 modes was 1.05, and the mean of the ratios $R_{max(EES/PTEEM)}$ over the 20 modes was 1.13).

Figures 1 and 2 show the last 2500 iterations after burn-in for the chains 1, 7, 14 and 20 obtained by one run of the PT algorithm, and by one run of the PTEEM algorithm. Figure 3 shows the simulations after a burn-in period for chains 1 to 5 obtained by a run of EES. The first chains of the PTEEM and EES visited all the modes of the target density whereas the first chain of PT did not visit all of them. Notice that chains with the highest temperatures of the PT algorithm visited all the modes, and these chains for the EES kept in memory lots of iterations.

Table 6 presents the repartition of accepted equi-energy moves for chains 1,10 and 20, with other possible chains within a run of the PTEEM algorithm. As expected, the closer the temperatures of chains were, the more often the equi-energy moves were accepted. Note that equi-energy moves had been proposed and accepted for all possible pairs of chains, including for pairs of chains with very different temperatures.

As in Kou et al. [2006], it appeared that the EES algorithm gave better results than the classical PT. Besides the PTEEM algorithm gave results comparable to those of the EES, and even slightly better.

# 4 Comparisons in the case of local Gibbs samplers moves

Some difficulties are encountered to combine EES with a Gibbs sampler. Indeed, it does not seem obvious to sample from the truncated joint posterior distribution. Hence in the case where the chains are locally updated by a Gibbs sampler we compared only results of PT and PTEEM algorithms. Two illustrations of Gaussian mixtures are treated: an example with simulated data, and the well-known example of the Galaxy dataset.
For both illustrations we consider independent observations $y_1, \cdots, y_n$ from $k$ mixture components

$$y_i \sim \sum_{j=1}^{k} w_j f(.|\mu_j, \sigma_j^2), \qquad i = 1, \ldots, n,$$

with $k$ fixed and known and where $f(.|\mu_j, \sigma_j^2)$ denotes the density of the Gaussian distribution $\mathcal{N}(\mu_j, \sigma_j^2)$. The sizes of the $k$ groups are proportional to $w_1, w_2, \ldots, w_k$, which are the weights of the components. The parameters to be estimated are the means $\mu_j$, the variances $\sigma_j^2$, and

the weights $w_j$, for $j = 1, \ldots, k$.
The label of the component from which each observation is drawn is unknown, and a label vector $c$ which is a latent allocation vector is introduced as follows: $c_i = j$ if the observation $y_i$ is drawn from the $j^{th}$ component. The variables $c_i$ are supposed independent with distributions

$$p(c_i = j) = w_j, \qquad j = 1, \ldots, k.$$

Write $y = (y_i)_{i=1,\ldots,n}$, $\mu = (\mu_j)_{j=1,\ldots,k}$, $\sigma^2 = (\sigma_j^2)_{j=1,\ldots,k}$, $w = (w_j)_{j=1,\ldots,k}$ and $c = (c_i)_{i=1,\ldots,n}$. The $\mu_j$ and $\sigma_j^{-2}$ are supposed to be independent with the following priors:

$$\mu_j \sim \mathcal{N}(\xi, \kappa^{-1}), \qquad \sigma_j^{-2} \sim \Gamma(\alpha, \beta) \qquad \text{and} \qquad \beta \sim \Gamma(g, h), \tag{2}$$

where $\beta$ and $h$ are rate parameters. The prior on $w$ is taken as a symmetric Dirichlet distribution

$$w \sim D(\delta, \delta, \ldots, \delta).$$

The parameters $\delta$, $\xi$, $\kappa$, $\alpha$, $g$ and $h$ are supposed to be fixed. Let us denote by $m_j = \sum_{i=1}^{n} \mathbb{1}_{c_i=j}$ the number of observations labeled by $j$.

The joint posterior density, the full conditional distributions and the formula of the acceptance rate for the equi-energy move are given in appendix A.

On the following examples, the estimates of the parameters obtained after labeling were quite good and similar for the PT and PTEEM algorithms. They were even comparable to those obtained with a classical Gibbs sampler. The major difference between these three algorithms was the ability to explore the parameter space: the Gibbs sampler found one mode of the mixture posterior and usually was staying only on this mode, while the PT and PTEEM algorithms succeeded to jump from one mode to another. Consequently, on the examples we focused on the label-switching phenomenon (see Jasra et al. [2005]), and not on the estimation of the parameters.

### 4.1 Simulated data

Following Jasra et al. [2005], a vector $y$ of length 100 was simulated from a mixture of four Gaussian distributions:

$$\frac{1}{4}\Big(\mathcal{N}(-3, 0.55^2) + \mathcal{N}(0, 0.55^2) + \mathcal{N}(3, 0.55^2) + \mathcal{N}(6, 0.55^2)\Big).$$

The number of components is $k = 4$, and we chose for the fixed parameters in (2): $\alpha = 2$, $\xi = \bar{y}$, $\delta = 1$, $\kappa = 1/R^2$, $g = 0.2$ and $h = 10/R^2$, where $R = max(y) - min(y)$. The algorithms PT and PTEEM were run 100 times, each run consisting of 4000 iterations after a burn-in period of 1000 iterations. We used 20 chains and 5 energy rings. As in the previous example, the PT algorithm used a swap between neighboring temperature chains for the exchange operation, and only one swap was proposed at each iteration.
Concerning the energy ladder, after a run of a classical Gibbs sampler with target density $\pi$, we

chose $H_1 = 470$ and $H_5 = 650$. Four energy rings were chosen with levels evenly spaced between 470 and 650 on a logarithmic scale, the fifth ring including states having an energy value higher than 650. The levels were then 470, 509.7, 552.7, 599.4 and 650.
Concerning the temperatures, after few runs of PT and PTEEM we noted that there was an energy gap between chains of temperatures lower than 1.18 and chains of temperatures higher than 1.28. Indeed, chains of temperatures lower than 1.18 had almost all their states in the first ring, while chains of temperatures higher than 1.28 had almost all their states in the last two rings. To overcome this problem, several temperatures were introduced between 1.18 and 1.28. The temperatures were then set by the following way: 4 temperatures between 1 and 1.17, 10 temperatures between 1.18 and 1.28 and 6 temperatures between 1.30 and 10, the temperatures being evenly spaced on a logarithmic scale, so we obtained 1, 1.05, 1.11, 1.17, 1.18, 1.19, 1.2, 1.21, 1.22, 1.23, 1.25, 1.26, 1.27, 1.28, 1.3, 1.96, 2.94, 4.42, 6.65 and 10.
Table 7 shows for several chains the distributions of states in the energy rings.

Clearly, the mixture posterior has $k! = 24$ symmetric modes and, in theory, for a very high number of iterations, the chain of interest should have visited all modes, with equal frequencies. When the chain goes from one mode to another, there is the so-called label-switching phenomenon (see Jasra et al. [2005]). Such a phenomenon is a useful convergence diagnostic to check if the chain of interest has explored all possible labelings of the parameters. To compare PT and PTEEM algorithms we considered for each run of each algorithm both the number and the frequency of visited modes by the target chain. Table 8 shows that on 100 runs of PTEEM the target chain visited more modes than on 100 runs of PT. Hence the label-switching phenomenon seems to occur more often during a run of PTEEM than during a run of PT. We also counted in each of the 100 runs for the two algorithms how many times the target chain visited each mode in the last 4000 iterations. The absolute frequency error is given by $err_i = |\hat{f}_i - 1/4!|$, where $\hat{f}_i$ is the sample frequency of the $i$th mode being visited ($i = 1, \ldots, 4!$). We then calculated the mean and median of this absolute frequency error over the 100 runs and the 4! modes. Absolute frequency errors were slightly lower for PTEEM with a mean (resp. a median) of 4,9% (reps. 4.2%), compared to 4.1% (resp. 4.1%) for PT.

We studied further the equi-energy moves of the algorithm PTEEM. In Table 9 it appears that exchange moves were more frequent between chains with similar temperatures. The mean acceptance rates of equi-energy moves for PTEEM and of exchange moves for PT were 53% and 61%, respectively. Note that we could code the PT algorithm so that exchange moves can be proposed between any two chains and not only between adjacent chains. But in this case the mean acceptance rate of an exchange move would be much lower. In comparison the PTEEM algorithm has the advantage to propose exchanges moves between chains not necessarily adjacent, but always having states of similar energy values.

### 4.2 Galaxy dataset

We used the well-known Galaxy dataset (see for instance Richardson and Green [1997]). The data consist of the velocities of 82 distant galaxies diverging from our own.
The number of components is $k = 6$, and we took for the fixed parameters in (2): $\alpha = 3$, $\xi = 20$, $\delta = 1$, $\kappa = 1/R^2$, $g = 0.2$ and $h = 10/R^2$, where $R = 10$. The algorithms PT and PTEEM were

run 100 times, each run consisting of 10000 iterations after a burn-in period of 2000 iterations. We used 20 chains and 5 energy rings. As in the previous example, the PT algorithm used a swap between neighboring temperature chains for the exchange operation, and only one swap was proposed at each iteration.
Concerning the energy ladder, after a run of a classical Gibbs sampler with target density $\pi$, we chose $H_1 = 460$ and $H_5 = 540$. Four energy rings were obtained with levels evenly spaced between $H_1$ and $H_5$ on a logarithmic scale, the fifth ring containing all states having an energy value higher than $H_5$. The levels obtained were 460, 478.8, 498.4, 518.8 and 540.
We chose $N = 20$ temperatures between 1 and 4, with their inverses evenly spaced. We get 1.00, 1.04, 1.09, 1.13, 1.19, 1.25, 1.31, 1.38, 1.46, 1.55, 1.65, 1.77, 1.90, 2.05, 2.24, 2.45, 2.71, 3.04, 3.45 and 4.00.
Table 10 presents for several chains the distributions of states in the energy rings.

As in the previous example on simulated data, Table 11 shows that on 100 runs of PTEEM the target chain visited more modes than on 100 runs of PT. Hence the label-switching phenomenon occured more often with PTEEM than with PT.

We also counted in each of the 100 runs for the two algorithms how many times the target chain visited each mode in the last 10000 iterations. As in the simulated data set example, the absolute frequency errors were calculated for each mode: absolute frequency errors were slightly lower for PTEEM with a mean (resp. a median) of 0.118% (resp. 0.099%), compared to 0.126% (resp. 0.099%) for PT.

As in the previous example, Table 12 shows that equi-energy moves were more frequent between chains with similar temperatures.

The mean acceptance rates of the equi-energy moves for PTEEM and of the exchange moves for PT were of 50% and 61% respectively. In conclusion, both datasets illustrated that the PTEEM algorithm performs a better exploration of the parameter space than the PT algorithm, while in mean less exchanges between chains were performed for the PTEEM algorithm, compared to the PT algorithm.

# 5 Discussion

In this paper a new algorithm combining PT and EES was proposed. Thanks to a relevant equi-energy move, the proposed PTEEM algorithm allows a better exploration of the parameter space than the PT algorithm, while ensuring the reversibility of the exchange moves. Therefore the generated Markov process theoretically converges to $\pi^*$, and the first chain generates samples corresponding to the distribution of interest $\pi$. From a practical point of view, a drawback of the PTEEM algorithm compared to the PT algorithm is that an energy ladder is needed, but we explained a simple and practical way to obtain a relevant ladder, which proved to be efficient. Compared to EES, less storage is needed since iterations from the past are not all kept in memory.

When using a Metropolis-Hastings algorithm, on the same example as Kou et al. [2006] the PTEEM algorithm gave better results than the PT algorithm, and results comparable to those of the EES, even slightly better. When using a Gibbs sampler, a major inconvenience of the EES algorithm is its difficulty to be applied, since it would necessitate to sample from

a posterior distribution with an energy truncation. Hence only PT and PTEEM algorithms were compared on a simulated dataset and on the Galaxy dataset. It was observed that the PTEEM algorithm had a better ability to explore the parameter space since the label-switching phenomenon occurred more often.
A direction for future research is to use the PTEEM algorithm within the case of mixture models with unknown number of components, adapting the method of Reversible Jump (Richardson and Green [1997]) with the principle of equi-energy moves.

## References


C. Andrieu, A. Jasra, A. Doucet, and P. Del Moral. Convergence of the equi-energy sampler. *ESAIM: Proceedings*, 19:1–5, 2007. doi: 10.1051/proc:071901.

C. Andrieu, A. Jasra, A. Doucet, and P. Del Moral. A note on convergence of the equi-energy sampler. *Stochastic Analysis and Applications*, 26(2):298–312, 2008. doi: 10.1080/07362990701857178.

Y.F. Atchadé and J.S. Liu. Discussion of equi-energy sampler by kou, zhou and wong. *The Annals of Statistics*, 34(4):1620–1628, 2006.

Y.F. Atchadé, G.O. Roberts, and .S. Rosenthal. Towards optimal scaling of metropolis-coupled markov chain monte carlo. *Statistics and Computing*, 2010.

K.B. Athreya, H. Doss, and J. Sethuraman. On the convergence of the markov chain simulation methodthe annals of statistics. *The Annals of Statistics*, 24(1):69–100, 1996.

G. Behrens, N. Friel, and M. Hurn. Tuning tempered transitions. *Unpublished manuscript*, 2009.

C.J. Geyer and E.A. Thompson. Annealing markov chain monte carlo with applications to ancestral inference. *Journal of the American Statistical Association*, 90:909–920, 1995.

P.J. Green and A. Mira. Delayed rejection in reversible jump metropolis-hastings. *Biometrika*, 88:1035–1053, 2001.

W.K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 88:1035–1053, 1970.

X Hua and S. Kou. Convergence of the equi-energy sampler and its application to the ising model. *Statistica Sinica*, In press, 2010.

A. Jasra, C.C. Holmes, and D.A. Stephens. Markov chain monte carlo methods and the label switching problem in bayesian mixture modeling. *Statistical Science*, 20(1):50–67, 2005.

A. Jasra, D.A. Stephens, and C.C. Holmes. Population-based reversible jump markov chain monte carlo. *Biometrika*, 94:787–807, 2007.

S.C. Kou, Q. Zhou, and W.H. Wong. Equi-energy sampler with application in statistical inference and statistical mechanics. *The Annals of Statistics*, 34(4):1581–1619, 2006.

F. Liang and W.H. Wong. Real-parameter evolutionary monte carlo with applications to bayesian mixture models. *Journal of the American Statistical Association*, 96:653–666, 2001.

N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E. Teller. Equations of state calculations by fast computing machines. *Journal of Chemical Physics*, 21(6):1087–1092, 1953.

R.M. Neal. Sampling from multimodal distributions using tempered transitions. *Statistics and computing*, 6:353–366, 1996.

S. Richardson and P.J. Green. On bayesian analysis of mixtures with an unknown number of components (with discussion). *Journal of the Royal Statistical Society B*, 59:731–792, 1997.

C. Robert and G. Casella. *Monte Carlo statistical methods, second edition.* Springer, 2004.

L. Tierney. Markov chains for exploring posterior distributions. *Annals of Statistics*, 22:1701–1762, 1994.

# A Formula used for the comparisons in case of a Gibbs sampler

## A.1 Joint posterior densities

Write $x = (\mu, \sigma^{-2}, w, c, \beta)$. The joint posterior density from which the parameters should be drawn is:

$$\begin{aligned} \pi(x) = p(\mu, \sigma^{-2}, w, c, \beta \mid y) &\propto p(y \mid \mu, \sigma^{-2}, c, \beta, w) p(\mu, \sigma^{-2}, c, \beta, w), \\ &\propto p(y \mid \mu, \sigma^{-2}, c) p(\mu, \sigma^{-2}, c, \beta, w). \end{aligned}$$

Hence the $i$th chain should be drawn from

$$\pi_i(x) \propto \pi(x)^{\frac{1}{T_i}} \propto p(y \mid \mu, \sigma^{-2}, c)^{\frac{1}{T_i}} p(\mu, \sigma^{-2}, w, c, \beta)^{\frac{1}{T_i}}.$$

However, as noted by Jasra et al. [2007] and Behrens et al. [2009], tempering the whole posterior is problematic as there is no guarantee that the tempered posterior will remain proper. As a consequence, only the likelihood contribution is tempered and the priors are left untempered. The $i$th chain is then drawn from

$$\begin{aligned} \pi_i'(x) &\propto p(y \mid x)^{\frac{1}{T_i}} p(x), \\ &\propto p(y \mid \mu, \sigma^{-2}, c)^{\frac{1}{T_i}} p(\mu \mid \xi, \kappa^{-1}) p(\sigma^{-2} \mid \alpha, \beta) p(c \mid w) p(w \mid \delta) p(\beta \mid g, h). \end{aligned}$$

Given the following notations

$$\begin{aligned} x_i &= (\mu_i, \sigma_i^{-2}, w_i, c_i, \beta_i), \\ x_j &= (\mu_j, \sigma_j^{-2}, w_j, c_j, \beta_j), \\ \mu_i &= (\mu_{i1}, \mu_{i2}, \ldots, \mu_{ik}), \\ \sigma_i^{-2} &= (\sigma_{i1}^{-2}, \sigma_{i2}^{-2}, \ldots, \sigma_{ik}^{-2}), \\ w_i &= (w_{i1}, w_{i2}, \ldots, w_{ik}), \\ c_i &= (c_{i1}, c_{i2}, \ldots, c_{ik}), \\ m_i &= (m_{i1}, m_{i2}, \ldots, m_{ik}), \end{aligned}$$

we have

$$\begin{aligned} \pi_i'(x) &\propto p(y \mid \mu, \sigma^{-2}, c)^{\frac{1}{T_i}} p(\mu \mid \xi, \kappa^{-1}) p(\sigma^{-2} \mid \alpha, \beta) p(c \mid w) p(w \mid \delta) p(\beta \mid g, h) \\ &\propto \Big[\prod_{p=1}^{k} (\sigma_p \sqrt{(2\pi)})^{-m_p} exp\Big(-\sum_{l=1}^{n} \frac{(y_l - \mu_{c_l})^2}{2\sigma_{c_l}^2}\Big)\Big]^{\frac{1}{T_i}} \Big[\prod_{p=1}^{k} \frac{\kappa^{\frac{1}{2}}}{\sqrt{2\pi}} exp\Big(-\frac{1}{2}(\mu_p - \xi)^2 \kappa\Big)\Big] \\ &\quad \times \Big[\prod_{p=1}^{k} \frac{\beta^\alpha}{\Gamma(\alpha)} \sigma_p^{-2(\alpha-1)} exp(-\beta \sigma_p^{-2})\Big] \Big[\frac{n!}{\prod_{p=1}^{k} m_p!} \prod_{p=1}^{k} w_p^{m_p}\Big] \\ &\quad \times \Big[\frac{1}{B(\delta, \ldots, \delta)} \prod_{p=1}^{k} w_p^{\delta-1}\Big] \Big[\beta^{g-1} exp(-h\beta) \frac{hg}{\Gamma(g)}\Big], \end{aligned}$$

where

$$B(\delta, \ldots, \delta) = \frac{\prod_{p=1}^{k} \Gamma(\delta)}{\Gamma(\sum_{p=1}^{k} \delta)} = \frac{\Gamma(\delta)^k}{\Gamma(k\delta)}.$$

## A.2 Full conditional distributions

Concerning the $i$th chain, the full conditional distributions to be used in the Gibbs sampler of the algorithms are easily obtained through conjugacy. For $\mu_j$, $\sigma_j^{-2}$ and $w$ they are the following

$$\mu_j \mid \sigma_j^{-2}, y, c, \xi, \kappa^{-1} \sim \mathcal{N}\left(\Big(\frac{m_j \sigma_j^2}{T_i} + \kappa\Big)^{-1} \Big(\frac{\sigma_j^2}{T_i} \sum_{l:c_l=j} y_l + \xi\kappa\Big), \Big(\frac{m_j \sigma_j^2}{T_i} + \kappa\Big)^{-1}\right),$$

$$\sigma_j^{-2} \mid \mu_j, y, c, \alpha, \beta \sim \Gamma\Big(\alpha + \frac{m_j}{2T_i}, \beta + \sum_{l:c_l=j} \frac{(y_l - \mu_j)^2}{2T_i}\Big),$$

$$w \mid c, \delta \sim D(\delta + m_1, \delta + m_2, \ldots, \delta + m_k).$$

For the allocation vector $c$, the full conditional distribution is multinomial with the following probabilities:

$$p_i(c_l = j \mid y, \mu, \sigma^{-2}, w) \propto \frac{1}{\sigma_j^{\frac{1}{T_i}}} exp\Big(-\frac{(y_l - \mu_j)^2}{2\sigma_j^2 T_i}\Big) w_j.$$

Eventually the $\beta$ parameter has the following full conditional distribution:

$$\beta \mid \sigma^{-2}, \alpha, g, h \sim \Gamma\Big(g + k\alpha, h + \sum_{j=1}^{k} \sigma_j^{-2}\Big).$$

### A.3 Acceptance rate of an equi-energy move

Assuming that two chains $i$ and $j$ are selected from an energy ring to be swapped, the acceptance probability of an equi-energy move proposed between two chains is given by

$$\min\left(1, \frac{\pi_i'(x_j)\pi_j'(x_i)}{\pi_i'(x_i)\pi_j'(x_j)}\right) = \min\left(1, \Big(\frac{p(y|x_i)}{p(y|x_j)}\Big)^{(1/T_j - 1/T_i)}\right),$$

with

$$\frac{\pi_i'(x_j)\pi_j'(x_i)}{\pi_i'(x_i)\pi_j'(x_j)} = \left[\frac{\prod_{p=1}^{k} \sigma_{jp}^{-m_{jp}}}{\prod_{p=1}^{k} \sigma_{ip}^{-m_{ip}}}\right]^{\frac{1}{T_i} - \frac{1}{T_j}} e\left[-\frac{1}{2}\Big(\frac{1}{T_i} - \frac{1}{T_j}\Big)\Big(\sum_{l=1}^{n}(y_l - \mu_{jc_{jl}})^2\sigma_{jc_{jl}}^{-2} - \sum_{l=1}^{n}(y_l - \mu_{ic_{il}})^2\sigma_{ic_{il}}^{-2}\Big)\right].$$

## B Proofs of Proposition 2.1 and Lemma 2.1

### B.1 Proof of Proposition 2.1

During an iteration of the PTEEM algorithm all chains are locally updated by a MCMC algorithm and an exchange move is proposed.
As the $i$th chain is locally updated by a MCMC algorithm, $PL_i(.,.)$ is reversible with stationary distribution $\pi_i$.
It is then clear that $PL = \prod_{i=1}^{N} PL_i$ is also reversible. Let $A \in \mathcal{B}(\mathcal{X}^{\mathcal{N}})$, that can be written as $A_1 \times A_2 \times \ldots \times A_N$, with $A_i \in \mathcal{X}$, we have

$$\begin{aligned}
\pi^*(A) &= \prod_{i=1}^{N} \pi_i(A_i) \\
&= \prod_{i=1}^{N} \int_{\mathcal{X}} PL_i(x_i, A_i)\pi_i(dx_i) \\
&= \int_{\mathcal{X}} \ldots \int_{\mathcal{X}} PL_1(x_1, A_1) \ldots PL_N(x_N, A_N)\pi_1(dx_1) \ldots \pi_N(dx_N) \\
&= \int_{\mathcal{X}^N} PL(s, A)\pi^*(ds),
\end{aligned}$$

which implies that $\pi^*$ is the stationary distribution of $PL(.,.)$.

With $q(s, s')$ the auxiliary distribution to propose $s'$ from $s$ in the equi-energy move, the transition kernel $PE$ can be written as

$$PE(s, s') = q(s, s')\rho(s, s') + \int_{\mathcal{X}} q(s, s'')(1 - \rho(s, ds''))\mathbb{1}_{\{s'\}}(s). \tag{3}$$

The detailed balance condition is

$$\forall A, B \in \mathcal{X}^{N^2}, \quad \int_A \int_B PE(s, ds')\pi^*(ds) = \int_B \int_A PE(s', ds)\pi^*(ds'). \tag{4}$$

From (3) we have

$$\forall B \in \mathcal{X}^N, \quad PE(s, B) = \int_B q(s, ds')\rho(s, s') + \int_{\mathcal{X}} q(s, ds'')(1 - \rho(s, s''))\mathbb{1}_B(s),$$

hence

$$\int_A \pi^*(ds) \int_B q(s, ds')\rho(s, s') + \int_{A\cap B} \pi^*(ds) \int_{\mathcal{X}} q(s, ds'')(1 - \rho(s, s'')) = \int_B \pi^*(ds') \int_A q(s', ds)\rho(s', s) + \int_{B\cap A} \pi^*(ds') \int_{\mathcal{X}} q(s', ds'')(1 - \rho(s', s'')).$$

Then (4) is satisfied if

$$\int_A \int_B \pi^*(ds)q(s, ds')\rho(s, s') = \int_B \int_A \pi^*(ds')q(s', ds)\rho(s', s),$$

which is established if the integrands are equal, that is if

$$q(s, ds')\rho(s, s')\pi^*(ds) = q(s', ds)\rho(s', s)\pi^*(ds'). \tag{5}$$

In PTEEM algorithm, the two candidate chains to exchange their actual states are chosen uniformly among all chains in the same energy ring. Hence we have $q(s, s') = q(s', s)$. Using (1), it follows that (5) is satisfied, and the detailed balance condition (4) holds. Therefore the transition kernel $PE$ for the equi-energy move is reversible, with stationary distribution $\pi^*$.

The transition kernels $PE$ and $PL$ are reversible with stationary distribution $\pi^*$. It is then clear that $P$ is also reversible. Moreover, we have

$$\begin{aligned}
\int_{\mathcal{X}^N} P(s, A)\pi^*(ds) &= \int_{\mathcal{X}^N} \int_{\mathcal{X}^N} PE(\tilde{s}, A)PL(s, \tilde{s})\pi^*(s)ds d\tilde{s} \\
&= \int_{\mathcal{X}^N} PE(\tilde{s}, A)\Big[\int_{\mathcal{X}^N} PL(s, \tilde{s})\pi^*(s)ds\Big]d\tilde{s} \\
&= \int_{\mathcal{X}^N} PE(\tilde{s}, A)\pi^*(\tilde{s})d\tilde{s} \\
&= \pi^*(A).
\end{aligned}$$

It follows that $\pi^*$ is the stationary distribution of $P$.

In addition, each $PL_i$ is supposed to be $\pi_i$-irreducible and aperiodic, hence $PL$ is aperiodic and $\pi^*$-irreducible. Since $PE$ is just an exchange kernel between two actual states it is clear that $P = PE * PL$ is also $\pi^*$-irreducible and aperiodic.

As $P$ is $\pi^*$-irreducible and aperiodic with stationary distribution $\pi^*$, following Theorem 1 of Tierney Tierney [1994], $S$ converges to its stationary distribution $\pi^*$ according to the total variation distance, for $\pi^*$-almost all $s \in \mathcal{X}^N$. Finally, as the marginal density of the first chain is $\pi_1 = \pi$, it provides samples corresponding to the target distribution.

□

## B.2 Proof of Lemma 2.1

From the assumption $\forall (x_i, x_i') \in \mathcal{X} \times \mathcal{X}$, $q_i(x_i, x_i') > 0$, $i \in \{1, \ldots, N\}$, it is clear that all state $s' \in \mathcal{X}^N$ such that $\pi^*(s') > 0$ can be reached from $s$. $S$ is then $\pi^*$-irreducible.

From Proposition 2.1, $S$ is reversible with stationary distribution $\pi^*$. As $S$ is $\pi^*$-irreducible, $S$ is positive.

Let $S^{(n)}$ denote the variable giving the state of $S$ at iteration $n$. To show that $S$ is Harris-recurrent we use Theorem 2 of Tierney [1994] that characterizes Harris-recurrent chains as follows: a Markov chain is Harris-recurrent if and only if the only bounded functions $h$ satisfying

$$\mathbb{E}(h(S^{(n)})|s_0) = \mathbb{E}(h(S^{(1)})|s_0) = h(s_0), \qquad \forall n \in \mathbb{N}, \tag{6}$$

are the constant functions. Functions $h$ satisfying (6) are called *harmonic*. By assumption, $\forall (x_i, x_i') \in \mathcal{X} \times \mathcal{X}$, $q_i(x_i, x_i') > 0$. It follows that $P(s, s') > 0$, $\forall (s, s') \in \mathcal{X} \times \mathcal{X}$. Indeed, let $s''$ corresponding to $s'$ for which the actual states of two chains with similar energy levels are switched. We have $\forall i \in \{1, \ldots, N\}$ $q_i(x_i, x_i'') > 0$, hence there is a non null probability to propose $s''$ from $s$. There is also a non null probability that all local moves are accepted, and therefore that $s''$ is reached from $s$ after all the $N$ local moves. Besides, there is a non null probability to propose $s'$ from $s''$ during the equi-energy move, as well as the probability to accept this equi-energy move is non null. Finally, there is a non null probability to reach $s'$ from $s$ in only one iteration of PTEEM. Following the same reasoning, it is clear that $\mathcal{X}$ is accessible. Then both assumptions (i) and (ii) of Theorem 6.80 of Robert and Casella [2004] are satisfied for $\mathcal{X}$, and this theorem (inspired from Athreya et al. [1996]) can be applied. It follows that if there exists $h$ satisfying (6), $h$ is $\pi^*$-almost everywhere constant and equal to $\mathbb{E}_{\pi^*}(h(S))$. We now show that $h$ is everywhere equal to $\mathbb{E}_{\pi^*}(h(S))$.
For all starting state $s_0 \in \mathcal{X}^N$, we have

$$\begin{aligned}
\mathbb{E}(h(S^{(1)})|s_0) &= \int_{\mathcal{X}^N} P(s_0, s_1) h(s_1) ds_1 \\
&= \int_{\mathcal{X}^N} \int_{\mathcal{X}^N} PE(\tilde{s}, s_1) PL(s_0, \tilde{s}) h(s_1) d\tilde{s} ds_1 \\
&= \int_{\mathcal{X}^N} \int_{\mathcal{X}^N} \Big( q(\tilde{s}; s_1) \rho(\tilde{s}; s_1) + (1 - r(\tilde{s})) \mathbb{1}_{s_1}(\tilde{s}) \Big) PL(s_0, \tilde{s}) h(s_1) d\tilde{s} ds_1 \\
&= \int_{\mathcal{X}^N} \Big( \int_{\mathcal{X}^N} q(\tilde{s}; s_1) \rho(\tilde{s}; s_1) h(s_1) ds_1 + (1 - r(\tilde{s})) h(\tilde{s}) \Big) PL(s_0, \tilde{s}) d\tilde{s},
\end{aligned}$$

with $r(s) = \int q(s, s') \rho(s, s') ds'$. Substituting $h(s_1)$ and $h(\tilde{s})$ by $\mathbb{E}_{\pi^*}(h(S))$ in the integral above

we obtain

$$\begin{aligned}
\mathbb{E}(h(S^{(1)})|s_0) &= \int_{\mathcal{X}^N} \Big( \int_{\mathcal{X}^N} q(\tilde{s}; s_1)\rho(\tilde{s}; s_1)\mathbb{E}_{\pi^*}(h(S))ds_1 + (1 - r(\tilde{s}))\mathbb{E}_{\pi^*}(h(S)) \Big) PL(s_0, \tilde{s})d\tilde{s} \\
&= \int_{\mathcal{X}^N} \Big( r(\tilde{s})\mathbb{E}_{\pi^*}(h(S)) + (1 - r(\tilde{s}))\mathbb{E}_{\pi^*}(h(S)) \Big) PL(s_0, \tilde{s})d\tilde{s} \\
&= \int_{\mathcal{X}^N} \mathbb{E}_{\pi^*}(h(S))PL(s_0, \tilde{s})d\tilde{s} = \mathbb{E}_{\pi^*}(h(S))
\end{aligned}$$

Because $\mathbb{E}(h(S^{(1)})|s_0) = h(s_0)$, it follows that $\forall s_0 \in \mathcal{X}^N$, $h(s_0) = \mathbb{E}_{\pi^*}(h(S))$. Then $h$ is a constant everywhere on $\mathcal{X}^N$, and $S$ is Harris-recurrent.

□

| | Bad repartition | | | | | Good repartition | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Energy ring | 1 | 2 | 3 | 4 | 5 | 1 | 2 | 3 | 4 | 5 |
| chain $i-2$ | 990 | 10 | 0 | 0 | 0 | 990 | 10 | 0 | 0 | 0 |
| chain $i-1$ | 950 | 50 | 0 | 0 | 0 | 701 | 202 | 97 | 0 | 0 |
| chain $i$ | 900 | 100 | 0 | 0 | 0 | 387 | 408 | 205 | 0 | 0 |
| chain $i+1$ | 0 | 2 | 237 | 511 | 250 | 45 | 312 | 355 | 288 | 0 |
| chain $i+2$ | 0 | 0 | 105 | 610 | 285 | 0 | 64 | 517 | 353 | 66 |

Table 1: Illustration for bad and good repartitions of the states in the energy rings. There is an energy gap between chains $i$ and $i+1$ in the bad repartition case.

| | Local moves | Exchange moves |
|---|---|---|
| EES | 0.387 | 0.799 |
| PT | 0.337 | 0.905 |
| PTEEM | 0.333 | 0.822 |

Table 2: Mean acceptance rates for local moves and exchange moves on 100 runs, for EES, PT and PTEEM algorithms.

| | $E(X_1)$ | $E(X_2)$ | $E(X_1)^2$ | $E(X_2)^2$ |
|---|---|---|---|---|
| True value | 4.478 | 4.905 | 25.605 | 33.920 |
| EES | 4.448 (0.301) | 4.953 (0.458) | 25.229 (3.112) | 34.226 (4.507) |
| PT | 3.971 (0.809) | 4.137 (1.114) | 21.510 (7.741) | 27.510 (10.407) |
| PTEEM | 4.483 (0.324) | 4.912 (0.454) | 25.556 (3.366) | 33.889 (4.406) |

Table 3: Estimations of the mean vector $(E(X_1), E(X_2))$ and of the second moments $(E(X_1^2), E(X_2^2))$ using the samples generated from the target chain, obtained on 100 runs for EES, PT and PTEEM algorithms. The standard deviations are given between parentheses.

| PT | EES | PTEEM |
|---|---|---|
| 2 to 10 missed. | 1 missed for 4 runs. | 1 missed for 2 runs. |
| A mean of 5.69 missed. | 2 missed for 2 runs. | |

Table 4: Number of missed modes by the 100 runs for EES, PT and PTEEM algorithms.

| | $\mu_1$ | $\mu_2$ | $\mu_3$ | $\mu_4$ | $\mu_5$ | $\mu_6$ | $\mu_7$ | $\mu_8$ | $\mu_9$ | $\mu_{10}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| $R_{med}$ PT/EES | 2.16 | 2.80 | 2.92 | 2.22 | 1.98 | 2.21 | 3.10 | 2.07 | 2.07 | 2.69 |
| $R_{max}$ PT/EES | 3.59 | 2.61 | 2.81 | 2.10 | 1.55 | 2.43 | 2.54 | 1.53 | 2.93 | 4.50 |
| $R_{med}$ PT/PTEEM | 2.60 | 3.72 | 2.63 | 2.19 | 1.79 | 2.97 | 2.77 | 2.55 | 2.32 | 2.64 |
| $R_{max}$ PT/PTEEM | 3.44 | 1.76 | 2.27 | 2.30 | 2.44 | 3.06 | 5.23 | 2.92 | 2.83 | 5.23 |
| $R_{med}$ EES/PTEEM | 1.21 | 1.33 | 0.90 | 0.99 | 0.91 | 1.35 | 0.89 | 1.23 | 1.12 | 0.98 |
| $R_{max}$ EES/PTEEM | 0.96 | 0.67 | 0.81 | 1.09 | 1.58 | 1.26 | 2.06 | 1.91 | 0.96 | 1.16 |
| | $\mu_{11}$ | $\mu_{12}$ | $\mu_{13}$ | $\mu_{14}$ | $\mu_{15}$ | $\mu_{16}$ | $\mu_{17}$ | $\mu_{18}$ | $\mu_{19}$ | $\mu_{20}$ |
| $R_{med}$ PT/EES | 2.51 | 2.46 | 2.77 | 2.63 | 2.39 | 1.76 | 3.06 | 2.22 | 2.10 | 2.37 |
| $R_{max}$ PT/EES | 4.58 | 1.60 | 3.23 | 4.61 | 3.26 | 2.10 | 2.83 | 4.77 | 3.50 | 1.36 |
| $R_{med}$ PT/PTEEM | 2.14 | 1.98 | 1.79 | 2.84 | 2.75 | 2.18 | 2.72 | 2.78 | 2.43 | 2.60 |
| $R_{max}$ PT/PTEEM | 3.05 | 2.02 | 2.35 | 4.16 | 3.44 | 1.79 | 3.72 | 3.78 | 3.50 | 2.16 |
| $R_{med}$ EES/PTEEM | 0.85 | 0.81 | 0.65 | 1.08 | 1.15 | 1.24 | 0.89 | 1.25 | 1.16 | 1.10 |
| $R_{max}$ EES/PTEEM | 0.67 | 1.26 | 0.73 | 0.90 | 1.06 | 0.85 | 1.32 | 0.79 | 1.00 | 1.58 |

Table 5: For each mode, ratios of median ($R_{med}$) and ratios of maximum ($R_{max}$) are for PT over EES, PT over PTEEM, and EES over PTEEM. Each ratio is obtained on 100 runs.

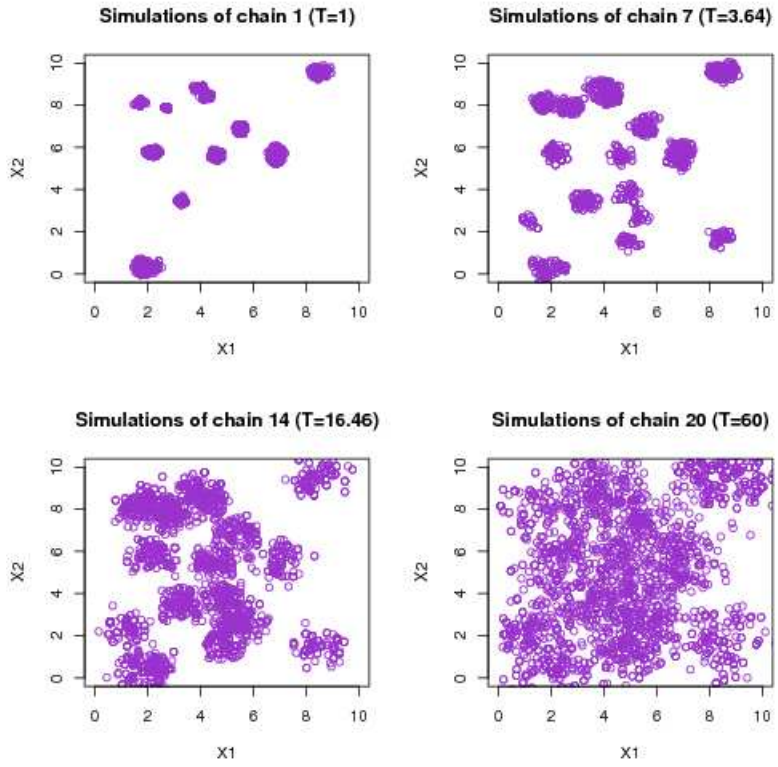


Figure 1: Simulations for chains 1, 7, 14 and 20 obtained by one run of the PT algorithm.

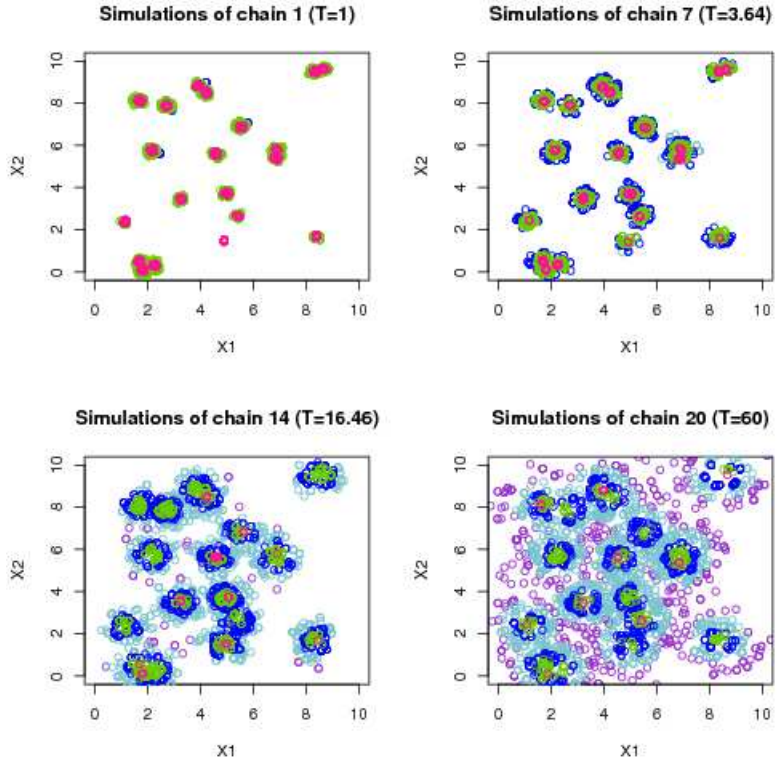


Figure 2: Simulations for chains 1, 7, 14 and 20 obtained by one run of the PTEEM algorithm. The colors correspond to the five energy levels.

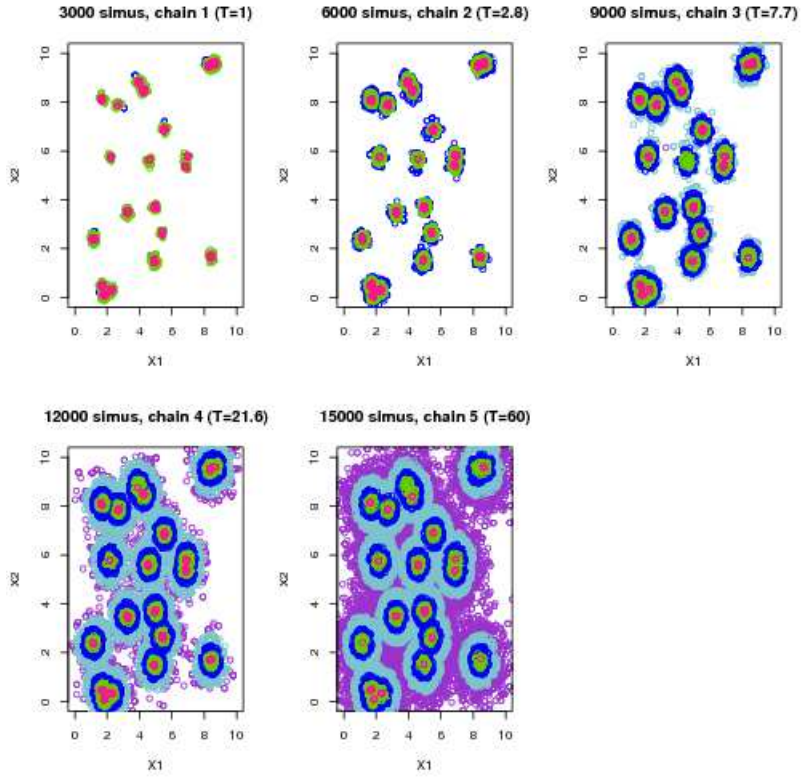


Figure 3: Simulations for chains 1 to 5 obtained by one run of the EES. The colors correspond to the five energy levels.

| | chain 1 | chain 10 | chain 20 |
|---|---|---|---|
| chain 1 | 0.00 | 4.63 | 0.50 |
| chain 2 | 16.32 | 4.33 | 0.62 |
| chain 3 | 14.34 | 4.29 | 0.64 |
| chain 4 | 11.98 | 4.64 | 0.70 |
| chain 5 | 9.96 | 4.89 | 0.76 |
| chain 6 | 8.26 | 5.46 | 1.03 |
| chain 7 | 6.57 | 5.76 | 1.17 |
| chain 8 | 6.01 | 6.26 | 1.50 |
| chain 9 | 4.96 | 6.74 | 2.03 |
| chain 10 | 4.32 | 0.00 | 2.30 |
| chain 11 | 3.25 | 7.11 | 3.13 |
| chain 12 | 2.85 | 6.67 | 4.33 |
| chain 13 | 2.42 | 6.65 | 5.61 |
| chain 14 | 1.98 | 6.11 | 7.38 |
| chain 15 | 1.61 | 5.76 | 8.75 |
| chain 16 | 1.44 | 5.13 | 10.64 |
| chain 17 | 1.15 | 4.64 | 13.72 |
| chain 18 | 1.08 | 4.32 | 16.09 |
| chain 19 | 0.87 | 3.63 | 19.10 |
| chain 20 | 0.62 | 2.99 | 0.00 |

Table 6: Repartition (in %) of accepted equi-energy moves between chain 1 and other possible chains (mean on 100 runs of PTEEM). Idem for chains 10 and 20.

| | $(-\infty, 509.7)$ | $[509.7, 552.7)$ | $[552.7, 599.4)$ | $[599.4, 650)$ | $[650, +\infty)$ |
|---|---|---|---|---|---|
| chain 1 | 4000 | 0 | 0 | 0 | 0 |
| chain 4 | 3557 | 333 | 109 | 1 | 0 |
| chain 8 | 2475 | 514 | 871 | 137 | 3 |
| chain 10 | 279 | 501 | 2036 | 1120 | 64 |
| chain 12 | 20 | 162 | 1483 | 2159 | 176 |
| chain 16 | 0 | 0 | 0 | 1769 | 2231 |
| chain 20 | 0 | 0 | 0 | 58 | 3942 |

Table 7: Distribution in the energy rings of states from 4000 iterations, for one run of PTEEM and for chains 1, 4, 8, 10 12, 16 and 20.

| | mean | standard deviation | min | max |
|---|---|---|---|---|
| PT | 12.42 | 1.77 | 8 | 16 |
| PTEEM | 16.54 | 2.1 | 11 | 20 |

Table 8: Means, standard deviations, minimal and maximal values of the number of visited modes, on 100 runs of PT and PTEEM.

| | chain 1 | chain 10 | chain 20 |
|---|---|---|---|
| chain 1 | 0.00 | 0.91 | 0.00 |
| chain 2 | 22.44 | 0.96 | 0.00 |
| chain 3 | 19.40 | 1.33 | 0.00 |
| chain 4 | 14.38 | 2.49 | 0.00 |
| chain 5 | 12.96 | 3.60 | 0.00 |
| chain 6 | 10.74 | 4.97 | 0.01 |
| chain 7 | 8.51 | 7.19 | 0.00 |
| chain 8 | 5.54 | 10.53 | 0.01 |
| chain 9 | 3.29 | 14.17 | 0.02 |
| chain 10 | 1.58 | 0.00 | 0.04 |
| chain 11 | 0.73 | 14.84 | 0.10 |
| chain 12 | 0.25 | 12.72 | 0.15 |
| chain 13 | 0.14 | 10.25 | 0.18 |
| chain 14 | 0.04 | 8.40 | 0.29 |
| chain 15 | 0.00 | 6.46 | 0.47 |
| chain 16 | 0.00 | 0.66 | 3.99 |
| chain 17 | 0.00 | 0.26 | 11.94 |
| chain 18 | 0.00 | 0.16 | 29.19 |
| chain 19 | 0.00 | 0.07 | 53.61 |
| chain 20 | 0.00 | 0.02 | 0.00 |

Table 9: Proportions (%) of accepted equi-energy moves between chain 1 and other possible chains (mean on 100 runs of PTEEM). Idem for chains 10 and 20.

| | $(-\infty, 478.8)$ | $[478.8, 498.4)$ | $[498.4, 518.8)$ | $[518.8, 540)$ | $[540, +\infty)$ |
|---|---|---|---|---|---|
| chain 1 | 9494 | 504 | 0 | 0 | 0 |
| chain 4 | 4136 | 5693 | 170 | 1 | 0 |
| chain 8 | 204 | 5872 | 3004 | 900 | 20 |
| chain 10 | 21 | 1573 | 3549 | 4273 | 584 |
| chain 12 | 0 | 65 | 919 | 5986 | 3030 |
| chain 16 | 0 | 0 | 19 | 1409 | 8572 |
| chain 20 | 0 | 0 | 0 | 154 | 9846 |

Table 10: Distribution in the energy rings of states from 10000 iterations, for one run of PTEEM and for chains 1, 4, 8, 10 12, 16 and 20.

| | mean | standard deviation | min | max |
|---|---|---|---|---|
| PT | 645.04 | 13.52 | 610 | 683 |
| PTEEM | 670.71 | 9.76 | 639 | 694 |

Table 11: Means, standard deviations, minimal and maximal values of the number of visited modes, on 100 runs of PT and PTEEM.

.

| | chain 1 | chain 10 | chain 20 |
|---|---|---|---|
| chain 1 | 0.00 | 0.02 | 0.00 |
| chain 2 | 64.02 | 0.09 | 0.00 |
| chain 3 | 23.65 | 0.28 | 0.00 |
| chain 4 | 7.77 | 0.82 | 0.00 |
| chain 5 | 2.67 | 1.63 | 0.00 |
| chain 6 | 1.11 | 2.84 | 0.00 |
| chain 7 | 0.47 | 5.88 | 0.00 |
| chain 8 | 0.22 | 11.85 | 0.01 |
| chain 9 | 0.07 | 21.28 | 0.05 |
| chain 10 | 0.02 | 0.00 | 0.24 |
| chain 11 | 0.00 | 23.34 | 0.67 |
| chain 12 | 0.00 | 14.96 | 1.61 |
| chain 13 | 0.00 | 8.53 | 3.26 |
| chain 14 | 0.00 | 4.30 | 5.82 |
| chain 15 | 0.00 | 2.05 | 8.69 |
| chain 16 | 0.00 | 0.98 | 12.61 |
| chain 17 | 0.00 | 0.57 | 17.14 |
| chain 18 | 0.00 | 0.28 | 22.13 |
| chain 19 | 0.00 | 0.19 | 27.76 |
| chain 20 | 0.00 | 0.11 | 0.00 |

Table 12: Proportions (%) of accepted equi-energy moves between chain 1 and other possible chains (mean on 100 runs of PTEEM). Idem for chains 10 and 20.