# ORTHOGONAL PARALLEL MCMC METHODS FOR SAMPLING AND OPTIMIZATION

*L. Martino*[⋆]*, V. Elvira*[†]*, D. Luengo*[‡]*, J. Corander*[◇]*, F. Louzada*[⋆]

[⋆] Institute of Mathematical Sciences and Computing, Universidade de São Paulo, São Carlos (Brazil).
[◇] Dep. of Mathematics and Statistics, University of Helsinki, Helsinki (Finland).
[†] Dep. of Signal Theory and Communic., Universidad Carlos III de Madrid, Leganés (Spain).
[‡] Dep. of Circuits and Systems Engineering, Universidad Politécnica de Madrid, Madrid (Spain).

## ABSTRACT

Monte Carlo (MC) methods are widely used in signal processing, machine learning and stochastic optimization. A well-known class of MC methods are Markov Chain Monte Carlo (MCMC) algorithms. In order to foster better exploration of the state space, specially in high-dimensional applications, several schemes employing multiple parallel MCMC chains have been recently introduced. In this work, we describe a novel parallel interacting MCMC scheme, called *orthogonal MCMC* (O-MCMC), where a set of "vertical" parallel MCMC chains share information using some "horizontal" MCMC techniques working on the entire population of current states. More specifically, the vertical chains are led by random-walk proposals, whereas the horizontal MCMC techniques employ independent proposals, thus allowing an efficient combination of global exploration and local approximation. The interaction is contained in these horizontal iterations. Within the analysis of different implementations of O-MCMC, novel schemes for reducing the overall computational cost of parallel multiple try Metropolis (MTM) chains are also presented. Furthermore, a modified version of O-MCMC for optimization is provided by considering parallel simulated annealing (SA) algorithms. We also discuss the application of O-MCMC in a big bata framework. Numerical results show the advantages of the proposed sampling scheme in terms of efficiency in the estimation, as well as robustness in terms of independence with respect to initial values and parameter choice.

***Index Terms—*** Parallel Markov Chain Monte Carlo (MCMC), Parallel Multiple Try Metropolis, Adaptive Multiple Try Metropolis, Block Independent Metropolis, Parallel Simulated Annealing.

## 1. INTRODUCTION

Monte Carlo (MC) methods are widely used in signal processing and communications [1, 2, 3]. Markov Chain Monte Carlo (MCMC) methods [4] are well-known MC methodologies to draw random samples and compute efficiently integrals involving a complicated multidimensional target probability density function (pdf), $\pi(\mathbf{x})$ with $\mathbf{x} \in \mathcal{D} \subseteq \mathbb{R}^{d_x}$. MCMC techniques only need to be able to evaluate the target pdf, but the difficulty of diagnosing and speeding up the convergence has driven intensive research effort in this field. For instance, several adaptive MCMC methods have been developed in order to determine adequately the shape and spread of the proposal density which is used to generate candidate samples, within the MCMC scheme [3, 5, 4, 6]. Nevertheless, guaranteeing the theoretical convergence is still an issue in most of the cases. Moreover, in a single specific (long) run, the generated chain can remain trapped in a local mode and, in this scenario, the adaptation could even slow down the convergence. Thus, in order to speed up the exploration of the state space (and specially to deal with high-dimensional applications [7]), several schemes employing parallel chains have been recently proposed [2, 6], as well as multiple try and interacting schemes [8], but the problem is still far from being solved. The interest in the parallel computation can be also originated by other motivations. For instance, several authors have studied the parallelization of MCMC algorithms which have traditionally been implemented in an iterative non-parallel fashion, in order to reduce their computation time [9, 10]. Furthermore, parallel MCMC schemes are required in a big data problems, where, for instance, the complete posterior distribution is split in different partial sub-posteriors [11, 12, 13, 14, 15].

In this work, we focus mainly on the implementation of parallel MCMC chains in order to foster the exploration of the state space and improve the overall performance.[1] More specifically, we present a novel family of parallel MCMC schemes,

[1] A preliminary version of this work has been published in [16]. With respect to that paper, here we propose different novel interacting schemes for exchanging information among the chains, analyze the theoretical basis of proposed approach and discuss certain relationships with other techniques, in detail. Several variants are presented for reducing the overall computational cost and for applying O-MCMC in optimization problems. Furthermore, we provide more exhaustive numerical simulations.

called orthogonal MCMC (O-MCMC) algorithms, where $N$ different chains are independently run and, at some iterations, they exchange information using another MCMC technique applied on the entire cloud of current states. Assuming that all the MCMC techniques used yield chains converging to the target pdf, the ergodicity property is guaranteed: the whole kernel is still valid, since it is obtained as the multiplication of ergodic kernels with the same invariant pdf. Fixing a given computational cost, this computing effort can be divided in $N$ parallel processes but, at some iteration, information among the chains is exchanged in order to enhance the overall mixing. The novel O-MCMC scheme is able to combine efficiently both the random-walk and the independent proposal approaches, as both strategies have advantages and drawbacks. On the one hand, random-walk proposal pdfs are often used when there is no specific information about the target, since this approach turns out to be more explorative than using a fixed proposal. On the other hand, a well-chosen independent proposal density usually provides less correlation among the samples in the generated chain. The novel method can mix both approaches efficiently: the parallel "vertical" chains (based on random-walk proposals) move around as "free explorers" roaming the state space, whereas the "horizontal" MCMC technique (applied over the population of current states and based on independent proposals) works as a "park ranger", redirecting "lost explorers" towards the "beaten track" according to the target pdf. Unlike in [17], the exchange of information occurs taking always into account the whole population of current states, instead of applying *crossover* or *exchange* schemes between specific pairs of chains. Furthermore, in this work, tempering of the target pdf is not considered for sampling purposes (but, it is employed for optimization). Hence, in this sense our approach resembles the nonreversible parallel MH algorithms described in [18, 19] where the whole population of states is also updated jointly at the times of interaction, pursuing non-reversibility instead of tempering as a means to accelerate convergence towards posterior mode regions. However, both tempering and crossovers could also be easily implemented within an O-MCMC framework.

Another contribution of the work is the computational improvement provided by novel parallel implementations of MCMC techniques using multiple candidates at each iteration. We present two novel schemes for parallel Multiple try Metropolis (MTM) [20, 21, 22] chains in order to reduce the overall computational cost in the same fashion of [9], saving generated samples, target evaluations and multinomial sampling steps. One of them is an extended version, using several candidates, of the Block Independent Metropolis presented in [9]. The ergodicity of both schemes is guaranteed. These novel parallel MTM techniques are employed as horizontal methods in O-MCMC. The corresponding O-MCMC scheme (using a novel parallel MTM method) can be also interpreted as an MTM algorithm employing an *adaptive* proposal density. This pdf is a mixture of $N$ components: the adaptation of the location parameters of the $N$ components is driven by the vertical parallel chains (note that the outputs of these chains are also used in the estimation). Furthermore, we describe a modified version of O-MCMC for solving optimization problems, considering parallel Simulated Annealing algorithms [23, 24, 25] for the vertical movements. The application for big data problems is also discussed. Numerical simulations show that O-MCMC exhibits both flexibility and robustness with respect to the initialization and parameterization of the proposals.

The paper is structured as follows. Section 2 summarizes the general framework and the aim of the work. Section 3 describes a generic complete O-MCMC scheme, whereas Sections 4 and 5 provide different specific examples of vertical and horizontal movements, respectively. Section 7 provides different numerical results. Finally, we finish with some remarks in Section 8.

## 2. BAYESIAN INFERENCE PROBLEM

In many applications, we aim at inferring a variable of interest given a set of observations or measurements. Let us denote the variable of interest by $\mathbf{x} \in \mathcal{D} \subseteq \mathbb{R}^{d_x}$, and let $\mathbf{y} \in \mathbb{R}^{d_y}$ be the observed data. The posterior pdf is then

$$\bar{\pi}(\mathbf{x}) \propto p(\mathbf{x}|\mathbf{y}) = \frac{\ell(\mathbf{y}|\mathbf{x})g(\mathbf{x})}{Z(\mathbf{y})}, \tag{1}$$

where $\ell(\mathbf{y}|\mathbf{x})$ is the likelihood function, $g(\mathbf{x})$ is the prior pdf and $Z(\mathbf{y})$ is the model evidence (a.k.a. marginal likelihood). In general, $Z(\mathbf{y})$ is unknown, so we consider the corresponding unnormalized target function,

$$\pi(\mathbf{x}) = \ell(\mathbf{y}|\mathbf{x})g(\mathbf{x}) \propto \bar{\pi}(\mathbf{x}). \tag{2}$$

In general, the study of the posterior density $\bar{\pi}(\mathbf{x})$ is impossible analytically (for instance, integrals involving $\bar{\pi}(\mathbf{x})$ are typically intractable), so that numerical approximations are required. Our goal is to approximate efficiently $\bar{\pi}(\mathbf{x})$ employing a cloud of random samples. In general, a direct method for drawing independent samples from $\bar{\pi}(\mathbf{x})$ is not available and alternative approaches (e.g., MCMC algorithms) are needed. The only required assumption is to be able of evaluating the unnormalized target function $\pi(\mathbf{x})$.

## 3. O-MCMC ALGORITHMS: GENERAL OUTLINE

Lt us consider $N$ parallel "vertical" chains, $\{\mathbf{x}_{n,t}\}_{n=1}^{N}$ with $t \in \mathbb{N}$, generated by different MCMC techniques with *random-walk* proposal pdfs $q_n(\mathbf{x}|\mathbf{x}_{i,t-1}) = q_n(\mathbf{x} - \mathbf{x}_{n,t-1})$, i.e., $\mathbf{x}_{n,t-1}$ plays the role of a location parameter for the density $q_n(\mathbf{x})$ in the next iteration. Let us denote the population of current states at the $t$-th iteration as

$$\mathcal{P}_t = \{\mathbf{x}_{1,t}, \mathbf{x}_{2,t}, \ldots, \mathbf{x}_{N,t}\}.$$

At certain selected iteration $t$, such that $t = mT_V$ where $T_V$ is a constant and $m = 1, \ldots, M$, we apply another MCMC technique taking into account the entire population of state $\mathcal{P}_{t-1}$, yielding a new cloud of samples $\mathcal{P}_t$. In this "horizontal" transitions, the different chains share information. The horizontal MCMC technique uses an proposal pdf $\varphi(\mathbf{x})$, which is independent from the previous states differently from the random walk proposals employed in the vertical MCMC chains. The general O-MCMC approach is represented graphically in Figure 1 and summarized below:

1. **Initialization:** Set $t = 1$. Choose the $N$ initial states,

$$\mathcal{P}_0 = \{\mathbf{x}_{1,0}, \mathbf{x}_{2,0}, \ldots, \mathbf{x}_{N,0}\},$$

   the total number of iterations, $T$, and three positive integer values $M, T_V, T_H \in \mathbb{N}\backslash\{0\}$ such that $M(T_V + T_H) = T$.

2. **For *m=1,...,M*:**

   (a) **Vertical period:** For

$$t = (m-1)(T_V + T_H) + 1, \ldots, mT_V + (m-1)T_H,$$

   run $N$ independent MCMC techniques starting from $\mathbf{x}_{n,t-1} \in \mathcal{P}_{t-1}$, thus obtaining $\mathbf{x}_{n,t}$, for $n = 1, \ldots, N$, and then a new population of states $\mathcal{P}_t = \{\mathbf{x}_{1,t}, \mathbf{x}_{2,t}, \ldots, \mathbf{x}_{N,t}\}$.

   (b) **Horizontal period:** For

$$t = mT_V + (m-1)T_H + 1, \ldots, m(T_V + T_H),$$

   apply an MCMC approach taking in account the entire population $\mathcal{P}_{t-1}$ at each step $t$, for generating the next cloud $\mathcal{P}_t$.

3. **Output:** Return the $T = M(T_V + T_H)$ samples contained in all the sets $\mathcal{P}_t$, $t = 1, \ldots, T$.

One vertical period contains $T_V$ iterations of the chains whereas, in one horizontal period we have $T_H$ iterations. Hence, given $t = (m-1)(T_V + T_H)$, after one cycle of vertical and horizontal steps we have $t = m(T_V + T_H)$. The total number of cycles (or epochs)[2] is $M = \frac{T}{T_V + T_H}$. The ergodicity is guaranteed if the vertical and horizontal steps produce ergodic chains with invariant density $\pi(\mathbf{x})$. See, Appendix A for further details. Relationships with other techniques, for instance [26, 27], are discussed in Section 5.4. In the following two sections, we discuss different specific possible implementations of O-MCMC.

## 4. VERTICAL MOVEMENTS

In this section, we describe different suitable implementations of the vertical parallel chains. Although it is not strictly necessary, we suggest using random walk proposal densities in the vertical chains. The idea is to exploit predominantly the explorative behaviors of the independent parallel MCMC methods. Therefore, we consider proposal of type $q_n(\mathbf{x}|\mathbf{x}_{n,t-1})$ where $\mathbf{x}_{n,t-1}$ plays the role of a location parameter. For instance, a sample $\mathbf{x}' \sim q_n(\mathbf{x}|\mathbf{x}_{n,t-1})$ can be expressed as

$$\mathbf{x}' = \mathbf{x}_{n,t-1} + \boldsymbol{\xi}_{n,t}, \tag{3}$$

where $\boldsymbol{\xi}_{n,t}$ has pdf $q(\boldsymbol{\xi})$ with zero mean and covariance matrix $\mathbf{C}_n$. Another more sophisticated possibility is to include the gradient information of the target within the proposal pdf as suggest in the *Metropolis-Adjusted Langevin Algorithm* (MALA) [28]. Namely, in this case a sample $\mathbf{x}' \sim q_n(\mathbf{x}|\mathbf{x}_{n,t-1})$,

$$\mathbf{x}' = \mathbf{x}_{n,t-1} + \frac{\epsilon}{2}\nabla \log\left[\pi(\mathbf{x}_{n,t-1})\right] + \boldsymbol{\xi}_{n,t}, \tag{4}$$

[2] One cycle, or epoch, includes one the vertical period and one horizontal period.

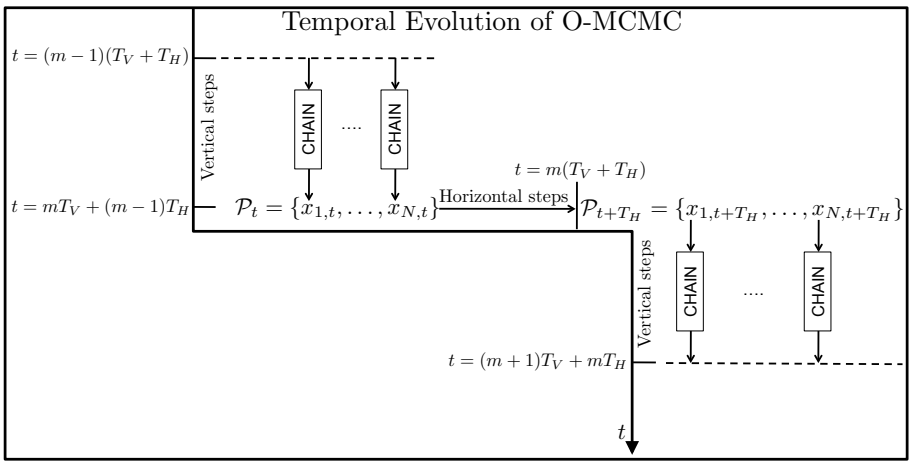



**Fig. 1**. A graphical representation of the O-MCMC approach. After $T_V$ vertical transitions, then $T_H$ horizontal steps are performed.

where $\boldsymbol{\xi}_{n,t}$ has pdf $q(\boldsymbol{\xi})$ with zero mean, covariance matrix $\mathbf{C}_n = \epsilon \mathbb{I}_{d_x \times d_x}$ and $\mathbb{I}_{d_x \times d_x}$ is the unit matrix and $\nabla f(\mathbf{x})$ denotes the gradient of a generic function $f(\mathbf{x})$. This second alternative can be particularly useful in high-dimensional spaces, although it inevitably increases the probability of the chain becoming trapped in one mode of the target, in a multi-modal scenario. Thus, the joint application of $N$ parallel chains appears very appropriate in this scenario. Moreover, the application of the O-MCMC scheme facilitates the jumps among different modes. Clearly, the random walk proposal density $q_n(\mathbf{x}|\mathbf{x}_{n,t-1})$ can be applied within different MCMC kernels. The simplest possibility consists in using a *Metropolis-Hastings* (MH) algorithm [4]. For each $n = 1, \ldots, N$ and for a given time step $t$, one MH update of the $n$-th chain is obtained as

1. Draw $\mathbf{x}' \sim q_n(\mathbf{x}|\mathbf{x}_{n,t-1})$.
2. Set $\mathbf{x}_{n,t} = \mathbf{x}'$ with probability
$$\alpha_n = \min\left[1, \frac{\pi(\mathbf{x}')q_n(\mathbf{x}_{n,t-1}|\mathbf{x}')}{\pi(\mathbf{x}_{n,t-1})q_n(\mathbf{x}'|\mathbf{x}_{n,t-1})}\right],$$
otherwise, with probability $1 - \alpha_n$, set $\mathbf{x}_{n,t} = \mathbf{x}_{n,t-1}$.

Other possible schemes can be used as alternative to MH: for instance, two famous alternatives are the *Multiple Try Metropolis* (MTM) [20, 22] and the *Delayed Rejection Metropolis* [29] techniques.

## 5. HORIZONTAL MOVEMENTS

As described above, at each iteration $t$, the vertical chains return a population $\mathcal{P}_t = \{\mathbf{x}_{1,t}, \ldots, \mathbf{x}_{N,t}\}$. When $t = mT_V + (m-1)T_H$, with $m \in \{1, ..., M\}$, i.e., after $T_V$ vertical transitions, then $T_H$ horizontal steps are performed. Here, we consider a generalized target density,

$$\bar{\pi}_g(\mathbf{x}_1, \ldots, \mathbf{x}_N) \propto \prod_{n=1}^{N} \pi(\mathbf{x}_n), \tag{5}$$

where each marginal, $\pi(\mathbf{x}_n)$ with $n = 1, ..., N$ and $\mathbf{x}_n \in \mathcal{D} \subseteq \mathbb{R}^{d_x}$, coincides with the target pdf in Eq. (2). The horizontal MCMC transitions leave invariant the extended target $\bar{\pi}_g$. Namely, after a "burn-in" period, the samples in the population $\mathcal{P}_t = \{\mathbf{x}_{1,t}, \ldots, \mathbf{x}_{N,t}\}$ are distributed according to $\bar{\pi}_g$. The purpose of these horizontal MCMC transitions is to share information among the $N$ different chains, improving the global mixing. The simplest possible horizontal scheme consists in employing an MH method directly in the extended domain, $\mathcal{D}^N \subseteq \mathbb{R}^{Nd_x}$, considering $\bar{\pi}_g$ as target. However, the probability of accepting a jump becomes negligible as $N$ grows, such as scheme becomes useless when a large number $N$ of vertical chains are used. In the following, we consider two different general approaches for sharing the information among the chains:

- In the first one, a population-based MCMC algorithm is applied. The states of vertical chains contained in $\mathcal{P}_t$ are used as initial population. Furthermore, the population-based MCMC scheme takes in account all the current population for

**Table 1**. Sample Metropolis-Hastings (SMH) algorithm for horizontal transitions in O-MCMC.

1. For $t = mT_V + (m-1)T_H + 1, \ldots, m(T_V + T_H)$:
   (a) Draw $\mathbf{x}_{0,t-1} \sim \varphi(\mathbf{x})$.
   (b) Choose a "bad" sample $\mathbf{x}_{k,t-1}$ in $\mathcal{P}_{t-1}$, i.e., $k \in \{1, ..., N\}$, according to the *inverse* of the importance sampling weights, i.e., with probability
   $$\gamma_k = \frac{\frac{\varphi(\mathbf{x}_{k,t-1})}{\pi(\mathbf{x}_{k,t-1})}}{\sum_{n}^{N} \frac{\varphi(\mathbf{x}_{n,t-1})}{\pi(\mathbf{x}_{n,t-1})}}, \qquad k = 1, \ldots, N.$$
   (c) Accept the new population,
   $$\mathcal{P}_t = \{\mathbf{x}_{1,t} = \mathbf{x}_{1,t-1}, \ldots, \mathbf{x}_{k,t} = \mathbf{x}_{0,t-1}, \ldots, \mathbf{x}_{N,t} = \mathbf{x}_{N,t-1}\},$$
   with probability
   $$\alpha = \frac{\sum_{n=1}^{N} \frac{\varphi(\mathbf{x}_{n,t-1})}{\pi(\mathbf{x}_{n,t-1})}}{\sum_{i=0}^{N} \frac{\varphi(\mathbf{x}_{i,t-1})}{\pi(\mathbf{x}_{i,t-1})} - \min\limits_{0 \leq i \leq N} \frac{\varphi(\mathbf{x}_{i,t-1})}{\pi(\mathbf{x}_{i,t-1})}}. \tag{6}$$
   Otherwise, set $\mathcal{P}_t = \mathcal{P}_{t-1}$.

making decisions about the next population. The simple case describes previously, i.e., the direct application of an MH scheme in the extended space, is an specific example of this approach.

- In the second one, the initial population $\mathcal{P}_t$ is also used for building a suitable proposal density $\psi(\mathbf{x})$. This pdf $\psi$ is employed by the $N$ parallel MCMC chains for yielding the next populations $\mathcal{P}_{t+1}, \ldots, \mathcal{P}_{t+T_H}$. More specifically, in this work, we suggest to construct $\psi(\mathbf{x})$ as a mixture of $N$ pdfs, each one centered in $\mathbf{x}_{n,t} \in \mathcal{P}_t$.

Next we show several specific examples. In all the different cases, for the horizontal movements, we consider the use of independent proposal pdfs, unlike for the vertical ones, where we have suggested the use of random walk proposals.

### 5.1. Population-based approach

The simplest possibility, in this case, is to apply a standard Metropolis-Hastings (MH) algorithm directly in the extended space $\mathbb{R}^{d_x \times N}$, i.e., with target $\bar{\pi}_g(\mathbf{x}_1, \ldots, \mathbf{x}_N)$, generating (block) transitions from $\mathcal{P}_{t-1} = \{\mathbf{x}_{1,t-1}, \ldots, \mathbf{x}_{N,t-1}\}$ to $\mathcal{P}_t = \{\mathbf{x}_{1,t}, \ldots, \mathbf{x}_{N,t}\}$. However, the probability of accepting a new population becomes dramatically lower as $N$ grows. An alternative example of possible population-based scheme, we describe the *Sample Metropolis-Hastings* (SMH) method [30, Chapter 4]. At each iteration, the underlying idea of SMH is replacing one "bad" sample in the population with a "better" one, according to a certain suitable probability. The new sample, candidate of be incorporated in the population, is generated from and independent proposal pdf $\varphi(\mathbf{x})$. The algorithm is designed so that, after a "burn-in" period $t_b$, the elements in $\mathcal{P}_{t'}$ ($t' > t_b$) are distributed according to $\bar{\pi}_g$ in Eq. (5). Table 1 shows the details of the algorithm.

Let us remark that the difference between $\mathcal{P}_{t-1}$ and $\mathcal{P}_t$ is at most one sample, and the acceptance probability, $0 \leq \alpha \leq 1$, depends on the entire population, $\mathbf{x}_{n,t-1}$ for $n = 1, \ldots, N$ and the proposed one, $\mathbf{x}_{0,t-1}$. At each step, the sample chosen to be replaced is selected according to a probability the inverse of the corresponding importance weight. The ergodicity can be proved considering the extended density $\bar{\pi}_g$ as target pdf. For further details see considerations in [31] and [30, Chapter 4]. Note also that the SMH algorithm becomes the standard MH method for $N = 1$. Hence, for $N = 1$ the specific O-MCMC implementation using SMH consists of applying alternatively two MH kernels with different types of proposals: a random walk proposal, $q_n(\mathbf{x}|\mathbf{x}_{n,t-1})$, and an independent one, $\varphi(\mathbf{x})$. This a well-known scheme (cf. [4, 30]), which can be seen as a particular case of the O-MCMC family of algorithms.

### 5.2. Mixture-based approach

An alternative approach consists in defining the following mixture of pdfs, updated each $T_V$ vertical transitions,

$$\psi(\mathbf{x}) = \psi_m(\mathbf{x}|\mathcal{P}_t) = \frac{1}{N} \sum_{n=1}^{N} \varphi_n(\mathbf{x}|\mathbf{x}_{n,t}), \quad \text{where} \quad t = mT_V + (m-1)T_H, \tag{7}$$

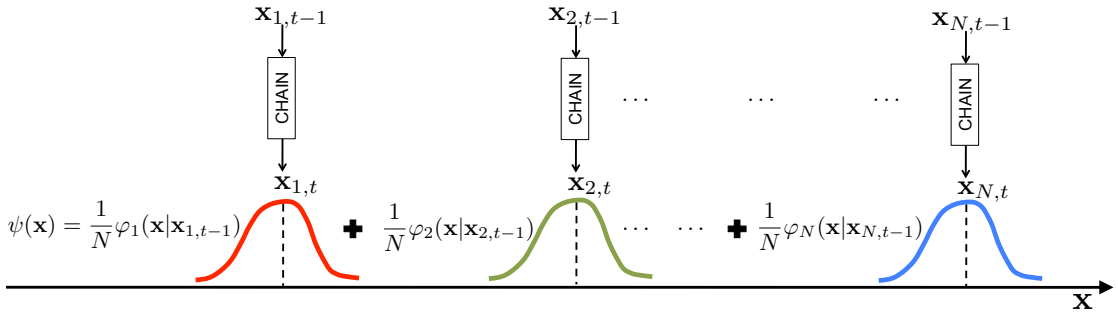



**Fig. 2**. A graphical representation of the mixture-based strategy. The mixture $\psi(\mathbf{x})$ is formed by $N$ components, $\varphi_n(\mathbf{x}|\mathbf{x}_{n,t})$, where $\mathbf{x}_{n,t} \in \mathcal{P}_t$ plays the role of location parameter.

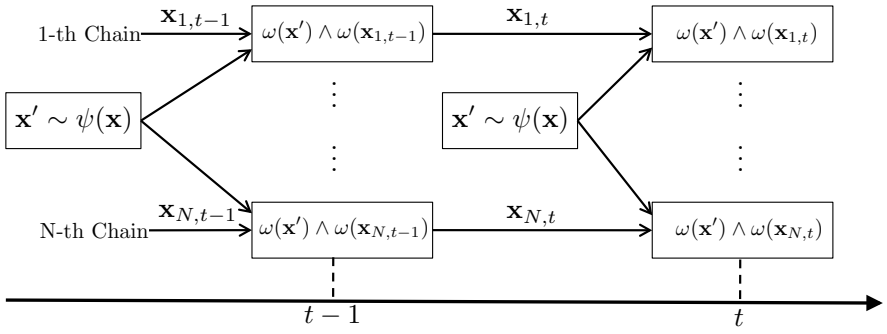



**Fig. 3**. A schematic representation of basic horizontal scheme described in Table 2. One specific transition of one specific chain is represented with the probability $\alpha_n = \omega(\mathbf{x}') \wedge \omega(\mathbf{x}_{n,t-1})$, showing the two possible future states at the $t$-th iteration, of the $n$-th chain.

where $m = 1, \ldots, M$ and each $\mathbf{x}_{n,t} \in \mathcal{P}_t$ plays the role of location parameter of the $n$-th component of the mixture, $\varphi_n$. Observe that $\psi(\mathbf{x})$ changes at the beginning of the horizontal iterations, considering the last vertical states, but remains fixed during horizontal iterations. Thus, during the complete O-MCMC run, we employ $M$ different mixtures $\psi$'s, one for each horizontal period, so that a more adequate notation would be $\psi_m(\mathbf{x})$ instead of $\psi(\mathbf{x})$. However, for simplifying the notation, we keep $\psi(\mathbf{x})$.

Figure 2 provides for a graphical representation. We employ $\psi(\mathbf{x})$ within $N$ independent MCMC schemes as an independent proposal density, namely independent from the previous state of the chain. The underlying idea is using the information in $\mathcal{P}_t$, with $t = mT_V + (m-1)T_H$, to build a good proposal function for performing $N$ independent MCMC processes. The theoretical motivation is that, after the burn-in periods, the vertical chains have converged to the target so that we can write $\mathbf{x}_{n,t} \sim \bar{\pi}(\mathbf{x})$ for $n = 1, \ldots, N$. Then, $\psi(\mathbf{x})$ in Eq. (7) can be interpreted as a kernel density estimation of $\bar{\pi}$ where $\varphi_n$ play the role of kernel functions.

As first example of this strategy, we consider the application of MH transitions. The procedure is shown in Table 2. At each iteration $t$, one sample $\mathbf{x}'$ is generated from $\psi(\mathbf{x})$ and then $N$ different MH tests are performed. Then, $T_H$ transitions of $N$ parallel chains are performed.

Alternatively, a different sample $\mathbf{x}'_n$, drawn from $\psi(\mathbf{x})$, can be tested for each chain, as shown in Table 3. Hence, $N$ different samples are drawn at each iterations (instead of only one) but, after building $\psi(\mathbf{x}|\mathcal{P}_t)$, the process could be completely parallelized. The variant in Table 3 provides in general better performance, although at the expense of a increasing computational cost, in terms of evaluations of the target and number of generated samples. However, the methodology called *block independent MH* [9], proposed exactly for reducing the computational effort recycling generated samples and target evaluations, can be also employed. For clarifying that, consider for simplicity $T_H = N$. Step 2(a) in Table 2 could be modified drawing $N$ independent samples $\mathbf{x}'_1, \ldots, \mathbf{x}'_N$ from $\psi(\mathbf{x})$ and, at each iteration $t$, a different circular permutation of the set $\{\mathbf{x}'_1, \ldots, \mathbf{x}'_N\}$ could be tested in the different $N$ acceptance MH tests[3]. Finally, observe that the procedure in Table 2 presents certain similarities with the Normal Kernel Coupler (NKC) method introduced in [32]. Clearly, NKC-type algorithms can be also employed as alternative

[3]For further clarifications, see the extension of this scheme for a Multiple Try Metropolis method described in Section 5.2.1.

**Table 2**. Basic mixture scheme for horizontal transitions in O-MCMC.

1. Build $\psi(\mathbf{x}) = \psi_m(\mathbf{x}|\mathcal{P}_t)$ as in Eq. (7) where $t = mT_V + (m-1)T_H$.
2. For $t = mT_V + (m-1)T_H + 1, \ldots, m(T_V + T_H)$:
   (a) Draw $\mathbf{x}' \sim \psi(\mathbf{x})$.
   (b) For $n = 1, \ldots, N$:
      i. Set $\mathbf{x}_{n,t} = \mathbf{x}'$, with probability
      $$\alpha_n = \min\left[1, \frac{\pi(\mathbf{x}')\psi(\mathbf{x}_{n,t-1})}{\pi(\mathbf{x}_{n,t-1})\psi(\mathbf{x}')}\right] = \omega(\mathbf{x}') \wedge \omega(\mathbf{x}_{n,t-1}).$$
      Otherwise, set $\mathbf{x}_{n,t} = \mathbf{x}_{n,t-1}$. We have denoted $\omega(\mathbf{x}) = \frac{\pi(\mathbf{x})}{\psi(\mathbf{x})}$ and $a \wedge b = \min[a, b]$, with $a, b \in \mathbb{R}$.
   (c) Set $\mathcal{P}_t = \{\mathbf{x}_{1,t}, \ldots, \mathbf{x}_{N,t}\}$.

**Table 3**. Variant of the basic mixture scheme for horizontal transitions in O-MCMC.

1. Build $\psi(\mathbf{x}) = \psi_m(\mathbf{x}|\mathcal{P}_t)$ as in Eq. (7) where $t = mT_V + (m-1)T_H$.
2. For $t = mT_V + (m-1)T_H + 1, \ldots, m(T_V + T_H)$:
   (a) For $n = 1, \ldots, N$:
      i. Draw $\mathbf{x}'_n \sim \psi(\mathbf{x})$.
      ii. Set $\mathbf{x}_{n,t} = \mathbf{x}'_n$, with probability
      $$\alpha_n = \min\left[1, \frac{\pi(\mathbf{x}'_n)\psi(\mathbf{x}_{n,t-1})}{\pi(\mathbf{x}_{n,t-1})\psi(\mathbf{x}'_n)}\right] = \omega(\mathbf{x}'_n) \wedge \omega(\mathbf{x}_{n,t-1}).$$
      Otherwise, set $\mathbf{x}_{n,t} = \mathbf{x}_{n,t-1}$. We have denoted $\omega(\mathbf{x}) = \frac{\pi(\mathbf{x})}{\psi(\mathbf{x})}$ and $a \wedge b = \min[a, b]$, with $a, b \in \mathbb{R}$.
   (b) Set $\mathcal{P}_t = \{\mathbf{x}_{1,t}, \ldots, \mathbf{x}_{N,t}\}$.

populated-based approaches.

More advanced techniques can be also modified and used as horizontal methods. Specifically, the adaptation to this scenario of multiple try schemes is particularly interesting. For instance, we adjust two special cases[4] of the *Ensemble MCMC* [33] and *Multiple Try Metropolis* methods [20, 34, 22] for fitting them within O-MCMC. Tables 4 and 5 summarize them. In both cases, $L \geq 1$ different i.i.d. samples $\mathbf{z}_1, \ldots, \mathbf{z}_L$ are draw from $\psi(\mathbf{x})$. In the parallel Ensemble MCMC (P-EnM) scheme, at each iteration $t$, one resampling step per chain is performed, considering the set of $L+1$ samples $\{\mathbf{z}_1, \ldots, \mathbf{z}_L, \mathbf{x}_{n,t-1}\}$, $n = 1, \ldots, N$ (using clearly importance weights). In the parallel MTM (P-MTM) scheme, at each iteration $t$, $N$ resampling steps are performed considering the set of $L$ candidates $\{\mathbf{z}_1, \ldots, \mathbf{z}_L\}$ and then the new possible states are tested (i.e., accepted or not) according to suitable acceptance probabilities $\alpha_n$, $n = 1, \ldots, N$, involving also the previous states $\mathbf{x}_{n,t-1}$.

The ergodicity of both schemes is discussed in Appendix C. The algorithms in Tables 4-5 are obtained by a rearrangement of the basic schemes in [33, 20, 34] in order to generate, at each iteration $t$, $N$ new states for $N$ independent parallel chains. The new states of the $N$ chains are selected filtering the same set of candidates $\{\mathbf{z}_1, \ldots, \mathbf{z}_L\}$, drawn from the same independent proposal pdf $\psi$. Note that, with respect to a *standard* parallel approach, they require less evaluations of the target pdf: at each iteration, the algorithms in Tables 4-5 require $L$ new evaluations of the target instead of $NL$ (as occurs in a standard parallel approach). For further explanations, see Appendix C.1.1 and Figure 5. With $L = 1$, the algorithm in Table 4 coincides with the application of $N$ parallel MH methods with Barker's acceptance rule [35]. The algorithm in Table 5 with $L = 1$ coincides with the scheme presented in Table 2. However, a number of tries $L \geq N$ is suggested.

### *5.2.1. Block Independent Multiple Try Metropolis algorithm*

Previously, we have pointed out that with the scheme in Table 5 only $L$ evaluations of the target are required at each iteration, instead of $NL$ as standard parallel approach. The proposed scheme in Table 5 can be also modified in the same fashion of the

[4] They are special cases of the corresponding algorithms, since an independent proposal pdf $\varphi$ is used.

**Table 4**. Parallel Ensemble MCMC (P-EnM) scheme for horizontal transitions in O-MCMC.

1. Build $\psi(\mathbf{x}) = \psi_m(\mathbf{x}|\mathcal{P}_t)$ as in Eq. (7) where $t = mT_V + (m-1)T_H$.
2. For $t = mT_V + (m-1)T_H + 1, \ldots, m(T_V + T_H)$:
   (a) Draw $L$ possible i.i.d. candidates $\mathbf{z}_1, \ldots, \mathbf{z}_L \sim \psi(\mathbf{x})$.
   (b) For $n = 1, \ldots, N$:
      i. Set $\mathbf{x}_{n,t} = \mathbf{z}_k \in \{\mathbf{z}_1, \ldots, \mathbf{z}_L\}$, i.e., with $k \in \{1, \ldots, L\}$, with probability

$$\alpha_k = \frac{\frac{\pi(\mathbf{z}_k)}{\psi(\mathbf{z}_k)}}{\sum_{\ell=1}^{L} \frac{\pi(\mathbf{z}_\ell)}{\psi(\mathbf{z}_\ell)} + \frac{\pi(\mathbf{x}_{n,t-1})}{\psi(\mathbf{x}_{n,t-1})}} \qquad k = 1, \ldots, L, \tag{8}$$

      or, set $\mathbf{x}_{n,t} = \mathbf{x}_{n,t-1}$ with probability

$$\alpha_{L+1} = 1 - \sum_{k=1}^{L} \alpha_k = \frac{\frac{\pi(\mathbf{x}_{n,t-1})}{\psi(\mathbf{x}_{n,t-1})}}{\sum_{\ell=1}^{L} \frac{\pi(\mathbf{z}_\ell)}{\psi(\mathbf{z}_\ell)} + \frac{\pi(\mathbf{x}_{n,t-1})}{\psi(\mathbf{x}_{n,t-1})}}. \tag{9}$$

      ii. Set $\mathcal{P}_t = \{\mathbf{x}_{1,t}, \ldots, \mathbf{x}_{N,t}\}$.

**Table 5**. Parallel Multiple Try Metropolis (P-MTM) scheme for horizontal transitions in O-MCMC.

1. Build $\psi(\mathbf{x}) = \psi_m(\mathbf{x}|\mathcal{P}_t)$ as in Eq. (7) where $t = mT_V + (m-1)T_H$.
2. For $t = mT_V + (m-1)T_H + 1, \ldots, m(T_V + T_H)$:
   (a) Draw $L$ possible i.i.d. candidates $\mathbf{z}_1, \ldots, \mathbf{z}_L \sim \psi(\mathbf{x})$.
   (b) Draw $N$ independent samples $\{\mathbf{z}_{k_1}, \ldots, \mathbf{z}_{k_N}\}$ such that $\mathbf{z}_{k_n} \in \{\mathbf{z}_1, \ldots, \mathbf{z}_L\}$, i.e., with $k_n \in \{1, \ldots, L\}$ and $n = 1, \ldots, N$, with probability

$$\beta_{k_n} = \frac{\frac{\pi(\mathbf{z}_{k_n})}{\psi(\mathbf{z}_{k_n})}}{\sum_{\ell=1}^{L} \frac{\pi(\mathbf{z}_\ell)}{\psi(\mathbf{z}_\ell)}}. \tag{10}$$

      Namely, resample $N$ times the samples in the set $\{\mathbf{z}_1, \ldots, \mathbf{z}_L\}$ with probability $\beta_k$, $k = 1, \ldots, L$.
   (c) For $n = 1, \ldots, N$:
      i. Set $\mathbf{x}_{n,t} = \mathbf{z}_{k_n}$ with probability

$$\alpha_n = \min\left[1, \frac{\sum_{\ell=1}^{L} \frac{\pi(\mathbf{z}_\ell)}{\psi(\mathbf{z}_\ell)}}{\sum_{\ell=1}^{L} \frac{\pi(\mathbf{z}_\ell)}{\psi(\mathbf{z}_\ell)} - \frac{\pi(\mathbf{z}_{k_n})}{\psi(\mathbf{z}_{k_n})} + \frac{\pi(\mathbf{x}_{n,t-1})}{\psi(\mathbf{x}_{n,t-1})}}\right]. \tag{11}$$

      Otherwise, set $\mathbf{x}_{n,t} = \mathbf{x}_{n,t-1}$ (with probability $1 - \alpha_n$).
   (d) Set $\mathcal{P}_t = \{\mathbf{x}_{1,t}, \ldots, \mathbf{x}_{N,t}\}$.

block independent MH method [9] in order to and reducing the number of multinomial sampling steps, without jeopardizing the ergodicity of the parallel chains. We remark that the corresponding technique, called *Block Independent Multiple Try Metropolis* (BI-MTM), can be always employed when $N$ parallel independent MTM are applied (even outside the O-MCMC scheme, clearly) in order to reduce the overall computational cost. Here, let us assume that the value $N$ is such that the number of total transitions of one chain, $T_H$, can be divided in $B = \frac{T_H}{N} \in \mathbb{N}$ blocks. The idea is based on using $N$ circular permutations of the resampled set $\{\mathbf{z}_{k_1}, \ldots, \mathbf{z}_{k_N}\}$, i.e.,

$$\begin{aligned}
\mathcal{V}_1 &= \{\mathbf{v}_{1,1} = \mathbf{z}_{k_1}, \mathbf{v}_{2,1} = \mathbf{z}_{k_2}, \ldots, \mathbf{v}_{N-1,1} = \mathbf{z}_{k_{N-1}}, \mathbf{v}_{N,1} = \mathbf{z}_{k_N}\}, \\
\mathcal{V}_2 &= \{\mathbf{v}_{1,2} = \mathbf{z}_{k_N}, \mathbf{v}_{2,2} = \mathbf{z}_{k_1}, \ldots, \mathbf{v}_{N-1,2} = \mathbf{z}_{k_{N-2}}, \mathbf{v}_{N,2} = \mathbf{z}_{k_{N-1}}\}, \\
&\vdots \\
\mathcal{V}_N &= \{\mathbf{v}_{1,N} = \mathbf{z}_{k_2}, \mathbf{v}_{2,N} = \mathbf{z}_{k_3} \ldots, \mathbf{v}_{N-1,N} = \mathbf{z}_{k_N}, \mathbf{v}_{N,N} = \mathbf{z}_{k_1}\},
\end{aligned} \tag{12}$$

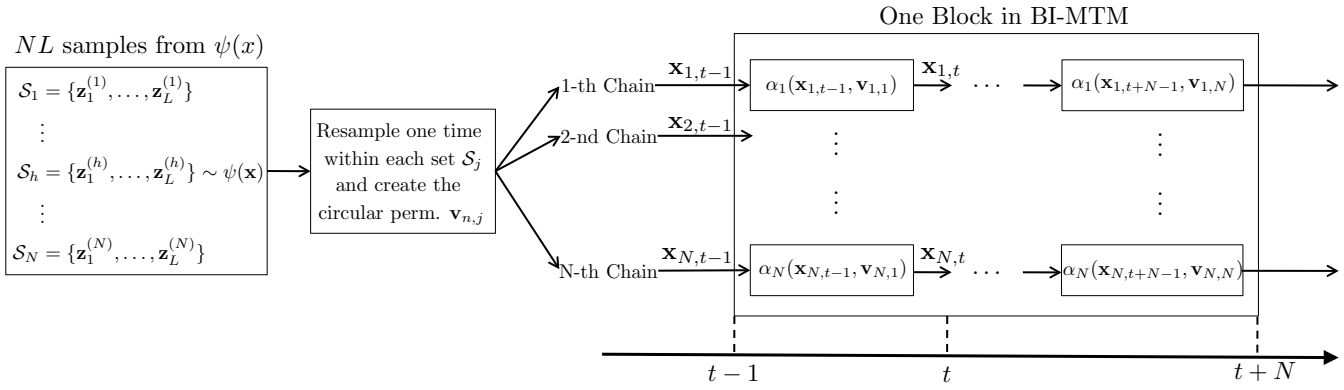



**Fig. 4**. A graphical representation of one block within the BI-MTM technique, described in Table 6. One specific transition of one MTM chain is represented with the probability $\alpha_n(\mathbf{x}_{n,t-1}, \mathbf{v}_{n,j})$, showing the two possible future states at the $t$-th iteration, of the $n$-th chain. One block is formed by $N$ transitions.

where each set $\mathcal{V}_n$ denotes one the $N$ possible circular permutations of $\{\mathbf{z}_{k_1}, \ldots, \mathbf{z}_{k_N}\}$. In order to preserve the ergodicity, each $\mathbf{z}_{k_j}$'s is drawn from a different set of tries $\mathcal{S}_j = \{\mathbf{z}_1^{(j)}, \ldots, \mathbf{z}_L^{(j)}\}$. More specifically, before a *block* of $N$ iterations, $NL$ tries are drawn from $\psi(\mathbf{x})$, yielding $N$ different sets $\mathcal{S}_j = \{\mathbf{z}_1^{(j)}, \ldots, \mathbf{z}_L^{(j)}\}$, $j = 1, \ldots, N$, each one containing $L$ elements. Then, one sample $\mathbf{z}_{k_j}$ is resampled from each $\mathcal{S}_j$ with probability proportional to the corresponding importance weight, and the circular permutations in Eq. (12) are created considering $\{\mathbf{z}_{k_1}, \ldots, \mathbf{z}_{k_N}\}$. The complete BI-MTM algorithm is detailed in Table 6 and further considerations are provided in Appendix C. In Table 6, we have denoted the acceptance probability as $\alpha_n(\mathbf{x}_{n,t-1}, \mathbf{v}_{n,j})$ for remarking the two possible future states of the $n$-th chain at the $t$-th iteration. Figure 4 depicts a schematic sketch of the different steps of one block within the BI-MTM algorithm. Moreover, Figure 5 provides a graphical comparison among different parallel MTM approaches. BI-MTM requires only $N$ multinomial sampling steps for each block, i.e., $N$ iterations, instead of $N^2$ as in P-MTM in Table 5. Moreover, BI-MTM is completely parallelizable. Indeed, alternatively one could draw $NLT_H$ samples from $\psi(\mathbf{x})$, perform $NT_H$ multinomial sampling steps within $NT_H$ different sets, and then run the $T_H$ parallel iterations of the $N$ chains, i.e., one unique block, using circular permutations of the $NT_H$ resampled samples (previously obtained). The reduction of computation cost is obtained at the expense of a moderate decrease of the performance.

### 5.3. Computational Cost

In general, the most costly steps are the evaluation of the target pdf, depending on the model or the number of data. The number of evaluations $E_H$ of the target, in one horizontal period, are $E_H = T_H$ for SMH in Table 1, whereas $E_H = LT_H$ in P-EnM and P-MTM (considering, in all cases, only the new evaluations at each iteration, the others can be automatically reused). Using SMH, $T_H$ multinomial sampling steps are performed, each one over a population of $N$ samples. In P-EnM and P-MTM, $NT_H$ multinomial sampling steps are required (with $N > 1$), each one over a set of $L$ samples. The total number of evaluations of the target $E_T = M(E_V + E_H)$ ,including the vertical transitions, is $E_T = M(NT_V + T_H)$ when the SMH is employed in the horizontal steps, or $E_T = M(NT_V + LT_H)$ when P-EnM and P-MTM are employed. Furthermore, in BI-MTM, we have again $E_T = M(NT_V + LT_H)$ but only $T_H$ multinomial sampling steps. Note also that in a standard parallel multiple try approach we would have $E_H = NLT_H$ evaluations of the target and $NT_H$ multinomial sampling steps, each one over a set of $L$ samples. Finally, we remark that, using SMH, we perform one acceptance test in step, i.e., $T_H$ in one horizontal period. Using a multiple candidates scheme, we employ $NT_H$ acceptance test in one horizontal period. All these considerations are summarized in Table 7. For further details and observations, see Appendix C.1.1.

### 5.4. Relationship with other techniques

The techniques in Table 4 and 5 are particular interesting since they involve the use of resampling steps without jeopardizing the ergodicity of the resulting global O-MCMC process. Moreover, the SMH algorithm in Table 1 employs an *inverted* resampling scheme since a sample in the population is chosen to be replaced with probability proportional to the inverse of the importance weights. Other methodologies in literature employ a combination of MCMC iterations and resampling steps, for instance, *iterated batch importance sampler* (IBIS) and *sequential Monte Carlo* (SMC) *samplers* for a static scenario [26, 27]. Their underlying idea could be interpreted belonging to the O-MCMC philosophy: in these methodologies, the resampling steps are

**Table 6**. Block Independent Multiple Try Metropolis (BI-MTM) algorithm for $N$ parallel chains.

1. Let $N$ be the total number of parallel MTM chains and $T_H$ be the total number of iterations of each chain, such that $\frac{T_H}{N} \in \mathbb{N}$. Choose a number of tries $L$. Set $t_0 = mT_V + (m-1)T_H$ if BI-MTM is used within O-MCMC, otherwise set $t_0 = 0$.
2. For each block $b = 1, \ldots, B = \frac{T_H}{N}$ do:
   (a) Draw $NL$ possible i.i.d. candidates $\mathbf{z}_1^{(h)}, \ldots, \mathbf{z}_L^{(h)} \sim \psi(\mathbf{x})$ with $h = 1, \ldots, N$.
   (b) Draw one sample $\mathbf{z}_{k_h}$ from each set $\mathcal{S}_h = \{\mathbf{z}_1^{(h)}, \ldots, \mathbf{z}_L^{(h)}\}$, $h = 1, \ldots, N$, with probability
   $$\beta_\ell^{(h)} = \frac{\frac{\pi(\mathbf{z}_\ell^{(h)})}{\psi(\mathbf{z}_\ell^{(h)})}}{\sum_{\ell=1}^{L} \frac{\pi(\mathbf{z}_\ell^{(h)})}{\psi(\mathbf{z}_\ell^{(h)})}}.$$
   Thus, finally we have $N$ different samples, $\{\mathbf{z}_{k_1}, \ldots, \mathbf{z}_{k_N}\}$, such that each $\mathbf{z}_{k_h} \in \mathcal{S}_h$.
   (c) Create the circular permutations $\mathbf{v}_{n,j} \in \{\mathbf{z}_{k_1}, \ldots, \mathbf{z}_{k_N}\}$ defined as in Eq. (12).
   (d) For $t = (b-1)N + 1 + t_0, \ldots, bN + t_0$ (i.e., exactly $N$ transitions):
      i. Set $j = t - (b-1)N - t_0$ (so that $j = 1, \ldots, N$, in one block).
      ii. For $n = 1, \ldots, N$:
         A. Set $\mathbf{x}_{n,t} = \mathbf{v}_{n,j}$, with probability
         $$\alpha_n(\mathbf{x}_{n,t-1}, \mathbf{v}_{n,j}) = \min\left[1, \frac{\sum_{\ell=1}^{L} \frac{\pi(\mathbf{z}_\ell^{(j)})}{\psi(\mathbf{z}_\ell^{(j)})}}{\sum_{\ell=1}^{L} \frac{\pi(\mathbf{z}_\ell^{(j)})}{\psi(\mathbf{z}_\ell^{(j)})} - \frac{\pi(\mathbf{v}_{n,j})}{\psi(\mathbf{v}_{n,j})} + \frac{\pi(\mathbf{x}_{n,t-1})}{\psi(\mathbf{x}_{n,t-1})}}\right].$$
         Otherwise, set $\mathbf{x}_{n,t} = \mathbf{x}_{n,t-1}$.
      iii. Set $\mathcal{P}_t = \{\mathbf{x}_{1,t}, \ldots, \mathbf{x}_{N,t}\}$.

**Table 7**. Computional cost of O-MCMC given different horizontal schemes. Recall that $M = \frac{T}{T_V + T_H}$.

| **Computational features** | **SMH** | **P-EnM and P-MTM** | **BI-MTM** | **Standard Parallel MTM chains** |
|---|---|---|---|---|
| $E_H$ | $T_H$ | $LT_H$ | $LT_H$ | $NLT_H$ |
| $E_T = M(E_V + E_H)$ | $M(NT_V + T_H)$ | $M(NT_V + LT_H)$ | $M(NT_V + LT_H)$ | $M(NT_V + NLT_H)$ |
| Total number of multinomial sampling steps | $MT_H$ | $MNT_H$ | $MT_H$ | $MNT_H$ |
| Cardinality of set for the multinomial sampling | $N$ | $L$ | $L$ | $L$ |
| Total number of acceptance tests | $M(NT_V + T_H)$ | $M(NT_V + NT_H)$ | $M(NT_V + NT_H)$ | $M(NT_V + NT_H)$ |

applied as an "horizontal" approach for interchanging information within the population. The resampling procedure generates samples from a particle approximation

$$\hat{\pi}^{(L)}(\mathbf{x}) = \sum_{\ell=1}^{L} \beta_\ell \delta(\mathbf{x} - \mathbf{z}_\ell), \tag{13}$$

of the measure of $\bar{\pi}(\mathbf{x})$, where $\mathbf{z}_\ell \sim \psi(\mathbf{x})$ (or, similarly, $\mathbf{z}_\ell \sim q_\ell(\mathbf{x})$ [36]) and $\beta_\ell$ are defined in Eq. (10) in Table 5, with $\ell = 1, \ldots, L$. The quality of this approximation improves as the number $L$ of samples grows. However, for a finite value of $L$ there exists a discrepancy which can produce problems in the corresponding sampling algorithm. For further details see Appendix B. One main issue is the loss in diversity in the population.

This problem is reduced drastically in O-MCMC since the ergodicity is ensured in both, vertical and horizontal movements. Clearly, this improvement in the performance is obtained at the expense of an increase of the computational cost. For instance, let us consider the use of SMH in horizontal transitions. The cloud of samples is not impoverished by the application of SMH, even if a poor choice of the proposal $\varphi(\mathbf{x})$ is made. In the worst case, the newly proposed samples are always discarded and computational time is wasted. In the best case, a proposal located in a low probability region can jump close to a mode of the

target. Moreover, in the the mixture approach, the mixture $\psi(\mathbf{x}) = \psi(\mathbf{x}|\mathcal{P}_t)$ is built using the states in $\mathcal{P}_t$ as location parameters, and then it does not change for the next $T_H$ horizontal steps. Thus, the information contained of the state $\{\mathbf{x}_{n,t}\}_{n=1}^N \in \mathcal{P}_t$ is employed in the next $T_H$ iterations even if some state is not well-located, fostering the safeguard of the diversity. For clarifying this point, consider for instance the basic scheme in Table 2. The mixture $\psi(\mathbf{x}) = \psi(\mathbf{x}|\mathcal{P}_t)$ does not change so that the information provided by the population $\mathcal{P}_t = \{\mathbf{x}_{1,t}, \ldots, \mathbf{x}_{N,t}\}$ at the iteration $t$ is still used in the iterations $t+1, \ldots, t+T_H$. Namely, using a "particle filtering slang", in these $T_H$ steps, "no particles are killed" where the "particles" in this case are the $N$ states $\{\mathbf{x}_{n,t}\}_{n=1}^N \in \mathcal{P}_t$ at the $t$-th iteration. This feature is also the main difference between the scheme in Table 2 and the NKC-type methods [32], where one component of the mixture is relocated after each iteration.

### 5.5. Joint adaptation of the proposal densities

Let us denote as $\mathbf{C}_n$ and $\mathbf{\Sigma}_n$ the scale parameters of the vertical and horizontal proposal pdfs, respectively. In order to design an algorithm as robust as possible, we suggest keeping the scale parameters $\mathbf{C}_n$ fixed for the vertical proposal pdfs $q_n(\mathbf{x}|\mathbf{x}_{n,t-1}, \mathbf{C}_n)$, to avoid a loss of diversity within the set of chosen variances. However, if desired, they could be adapted easily as suggested in [6]. On the other hand, we suggest of adapting the scale parameters of the horizontal proposal pdfs $\varphi_n$, $n = 1, \ldots, N$, since it is less delicate since a poor choice of the $\varphi_n$'s entails an increase in the computational cost but the diversity in the cloud of samples is always preserved. Several strategies have been proposed in [3, 37] and [6], for adapting proposal functions online within MCMC schemes. For the sake of simplicity, we discuss separately the cases of population-based or the mixture-based approaches.

- *Adaptation within SMH:* in this case, the strategies in [37, 6] are appropriate. Thus, After a training period $T_{train} < T$, *all* the generated samples (i.e., for each $t > T_{train}$ and from all the chains) can be used to adapt the location and scale parameters of proposal pdf $\varphi(\mathbf{x})$. Namely, denoting $\varphi_t(\mathbf{x}) = \varphi(\mathbf{x}; \boldsymbol{\mu}_t, \mathbf{\Sigma}_t)$, we can use the following approach:
  - If $t \leq T_{train}$: set $\boldsymbol{\mu}_t = \boldsymbol{\mu}_0$, $\mathbf{\Sigma}_t = \mathbf{\Sigma}_0$ (where $\boldsymbol{\mu}_0$ and $\mathbf{\Sigma}_0$ are the initial choices).
  - If $t > T_{train}$: set $\boldsymbol{\mu}_t = \frac{1}{Nt}\sum_{j=1}^t \sum_{n=1}^N \mathbf{x}_{n,j}$, and $\mathbf{\Sigma}_t = \frac{1}{Nt}\sum_{j=1}^t \sum_{n=1}^N (\mathbf{x}_{n,j} - \boldsymbol{\mu}_t)(\mathbf{x}_{n,j} - \boldsymbol{\mu}_t)^\top + \mathbf{C}$, where $\mathbf{C}$ is a chosen covariance matrix. The empirical mean and covariance matrix estimators can be also computed recursively [3].
- *Adaptation of the mixture $\psi(\mathbf{x})$:* the methods in Section 5.2 employ a mixture $\psi(\mathbf{x}) = \frac{1}{N}\sum_{n=1}^N \varphi_n(\mathbf{x})$. In this case, each components $\varphi_{n,t}(\mathbf{x}) = \varphi_{n,t}(\mathbf{x}; \boldsymbol{\mu}_t, \mathbf{\Sigma}_t)$ should be adapted, jointly with the weights of the mixture. A possible (and simple) adaptation scheme is provided in [3] where all the parameters of the mixture are updated online. The method in [3] can easily reformulated for a framework with parallel chains. In this case, the states of the parallel chains are divided in $N$ different clusters according to the euclidean distance between them and location parameters of the $N$ components in the mixture $\psi(\mathbf{x})$. Then, new centroids (i.e., location parameters), covariances matrices and weights are updated according to the mean, covariances and cardinality of each cluster, respectively.

## 6. O-MCMC FOR OPTIMIZATION AND BIG DATA CONTEXT

### 6.1. Interacting Parallel Simulated Annealing

We can be easily modified the O-MCMC schemes converting them in stochastic optimization algorithms. Indeed, it is possible replacing the $N$ vertical MH chains with $N$ parallel *simulated annealing* (SA) methods [23, 24]. Let us denote as $\gamma_{n,t} \in (0, +\infty)$ a finite scale parameter that is decreasing function of $t$ approaching zero for $t \to +\infty$, i.e.,

$$\begin{cases} \gamma_{n,t} \geq \gamma_{n,t+1} \geq \ldots \geq \gamma_{n,t+\tau} > 0, \\ \lim\limits_{t \to +\infty} \gamma_{n,t} = 0, \end{cases} \tag{14}$$

for $n = 1, \ldots, N$. Moreover, for the sake of simplicity, we consider symmetric proposal functions $q_n(\mathbf{y}|\mathbf{x}) = q_n(\mathbf{x}|\mathbf{y})$. Then, one transition of $n$-th SA is described below:

1. Draw $\mathbf{x}' \sim q_n(\mathbf{x}|\mathbf{x}_{n,t-1})$.
2. Set $\mathbf{x}_{n,t} = \mathbf{x}'$ with probability

$$\alpha_n = \min\left[1, \frac{[\pi(\mathbf{x}')]^{\frac{1}{\gamma_{n,t}}}}{[\pi(\mathbf{x}_{n,t-1})]^{\frac{1}{\gamma_{n,t}}}}\right] = \min\left[1, \left(\frac{\pi(\mathbf{x}')}{\pi(\mathbf{x}_{n,t-1})}\right)^{\frac{1}{\gamma_{n,t}}}\right],$$

otherwise, with probability $1-\alpha_n$, set $\mathbf{x}_{n,t}=\mathbf{x}_{n,t-1}$.

Above, with respect to the MH algorithm, we have replaced the target $\pi(\mathbf{x})>0$ with $[\pi(\mathbf{x})]^{\frac{1}{\gamma_{n,t}}}>0$ with modes that become sharper and narrower when we reduce the scale parameter $\gamma_{n,t}$. Note that the movements such $\pi(\mathbf{x}')>\pi(\mathbf{x}_{n,t-1})$ are always accepted whereas, when $\pi(\mathbf{x}')<\pi(\mathbf{x}_{n,t-1})$, they are accepted with probability $P_d=\left(\frac{\pi(\mathbf{x}')}{\pi(\mathbf{x}_{n,t-1})}\right)^{\frac{1}{\gamma_{n,t}}}\in(0,1]$. This probability $P_d\to 0$ vanishes to zero as $\gamma_{n,t}\to 0$ (guaranteeing the convergence to the global maximum when $t\to+\infty$). In the same fashion, for the horizontal steps we also consider the modified extended target,

$$\bar{\pi}_g(\mathbf{x}_1,\ldots,\mathbf{x}_N)\propto\prod_{n=1}^{N}[\pi(\mathbf{x}_n)]^{\frac{1}{\gamma_{n,t}}}, \tag{15}$$

so that all the presented schemes, previously described, can be automatically applied. Several possible decreasing functions $\gamma_{n,t}$ has been suggested in [23, 25, 24].

### 6.2. O-MCMC for big data and data-tempered distributions

In a Big Data context, the posterior density $\bar{\pi}(\mathbf{x})\propto\pi(\mathbf{x})$ is typically split in different partial posteriors, $\bar{\pi}_n(\mathbf{x})\propto\pi_n(\mathbf{x})$, with $n=1,\ldots,N$, each one considering a disjoint sub-sets of the observed data and such that

$$\bar{\pi}(\mathbf{x})\propto\prod_{n=1}^{N}\pi_n(\mathbf{x}),$$

The general idea is to generate samples from the partial posteriors $\bar{\pi}_n(\mathbf{x})$, and then to combine them in some way for approximating the complete target $\bar{\pi}(\mathbf{x})$ [12]. Thus, let us consider the application of $N$ parallel (vertical) MCMC chains, addressing one different partial target $\bar{\pi}_n(\mathbf{x})$. Since each observed data provides information about the same phenomenon, an interaction among the $N$ parallel chains can be improve the mixing of different MCMC techniques. In this context, the application of horizontal transitions using the mixture-based approach described in Section 5.2, appears particularly appropriate. Using the similar observations, O-MCMC can be apply within the so called "data point tempered" techniques [26] where a sequence of posteriors $\pi_1(\mathbf{x})$, $\pi_2(\mathbf{x})$,...,$\pi_N(\mathbf{x})$, with an increasing number of data, are considered (typically, the last one contains all the data, i.e., $\pi_N(\mathbf{x})=\pi(\mathbf{x})$).

## 7. NUMERICAL SIMULATIONS

### 7.1. Multimodal target distribution

In this section, we consider a bivariate multimodal target pdf, which is itself a mixture of 5 Gaussian pdfs, i.e.,

$$\bar{\pi}(\mathbf{x})=\pi(\mathbf{x})=\frac{1}{5}\sum_{i=1}^{5}\mathcal{N}(\mathbf{x};\boldsymbol{\nu}_i,\mathbf{G}_i),\quad \mathbf{x}\in\mathbb{R}^2, \tag{16}$$

with means $\boldsymbol{\nu}_1=[-10,-10]^\top$, $\boldsymbol{\nu}_2=[0,16]^\top$, $\boldsymbol{\nu}_3=[13,8]^\top$, $\boldsymbol{\nu}_4=[-9,7]^\top$, and $\boldsymbol{\nu}_5=[14,-14]^\top$, with covariance matrices $\mathbf{G}_1=[2,\ 0.6;0.6,\ 1]$, $\mathbf{G}_2=[2,\ -0.4;-0.4,\ 2]$, $\mathbf{G}_3=[2,\ 0.8;0.8,\ 2]$, $\mathbf{G}_4=[3,\ 0;0,\ 0.5]$, and $\mathbf{G}_5=[2,\ -0.1;-0.1,\ 2]$. With these values, the pdf $\pi(\mathbf{x})$ has 5 different modes. We apply O-MCMC to estimate the expected value $E[\mathbf{X}]$ of $\mathbf{X}\sim\bar{\pi}(\mathbf{x})$ (true value $E[\mathbf{X}]=[1.6,1.4]^\top$) using different values for the number of parallel chains $N\in\{5,100,1000\}$. Furthermore, we choose deliberately a "bad" initialization to test the robustness of the algorithm and its ability to improve the corresponding trivial parallel MH implementation. Specifically, we set $\mathbf{x}_{n,0}\sim\mathcal{U}([-4,4]\times[-4,4])$ for $n=1,\ldots,N$. This initialization is "bad" in the sense that it does not contain the modes of $\pi(\mathbf{x})$. In all cases, we consider $q_n(\mathbf{x}|\mathbf{x}_{n,t-1})=\mathcal{N}(\mathbf{x};\mathbf{x}_{n,t-1},\mathbf{C}_n)$ using the same isotropic covariance matrix, $\mathbf{C}_n=\sigma^2\mathbf{I}_2$, for every proposal of the vertical chains. We test different values of $\sigma\in\{2,5,10,70\}$ to gauge the performance of O-MCMC. In O-MCMC, we consider the application of SMH and P-MTM as horizontal techniques, as described below.

- *O-MCMC with SMH:* as horizontal proposal, we use again a Gaussian pdf, $\varphi_t(\mathbf{x})=\mathcal{N}(\mathbf{x};\boldsymbol{\mu}_t,\boldsymbol{\Sigma}_t)$ where $\boldsymbol{\mu}_t$ and $\boldsymbol{\Sigma}_t$ are adapted online as described in Section 5.5: namely, $\boldsymbol{\mu}_t=\frac{1}{Nt}\sum_{j=1}^{t}\sum_{n=1}^{N}\mathbf{x}_{n,j}$, and $\boldsymbol{\Sigma}_t=\frac{1}{Nt}\sum_{j=1}^{t}\sum_{n=1}^{N}(\mathbf{x}_{n,j}-$

$\boldsymbol{\mu}_t)(\mathbf{x}_{n,j}-\boldsymbol{\mu}_t)^\top+\boldsymbol{\Sigma}_0$, where $\boldsymbol{\mu}_0=[0,0]^\top$, $\boldsymbol{\Sigma}_0=\lambda^2\mathbf{I}_2$ with $\lambda=2$.[5] As remarked in Section 5.5, this adaptive procedure is quite robust since employs samples generated by different parallel chains [6]. Furthermore, we fix $T=4000$ and $T_H=T_V$. We test different values of $T_V\in\{1,100\}$ and, as a consequence, $M=\frac{T}{T_V+T_H}=\frac{T}{2T_V}\in\{20,2000\}$.[6] Recall that the total number of evaluations of the targets in O-MCMC with SMH is $E_T=M(NT_V+T_H)=\frac{T}{2}(N+1)$ (see Section 5.3).

- *O-MCMC with P-MTM:* WORK IN PROGRESS (see next versions in vixra.org)

We compare the performance of O-MCMC with the application of independent parallel chains (IPCs), namely, only vertical independent transitions. Therefore, we can infer the benefit of applying the horizontal interaction. For a fair comparison, in IPCs we use the same MH kernels, i.e., with the same proposals $q_n$'s, and we keep fixed the total number of evaluations of the target $E_T$ in both cases, O-MCMC and IPCs. Note that $E_T=NT'$ in IPCs where $N$ is the number of chains and $T'$ the total number of iterations for each one. We test $N\in\{5,100,1000\}$.

Table 8 shows the mean absolute error (MAE) in the estimation of the first component of the mean averaged over 1000 independent runs. O-MCMC with SMH always outperforms IPCs, specially for small $\sigma$ and $N$. O-MCMC shows a much more stable behavior w.r.t. the parameter choice $\sigma$. For large scale parameters ($\sigma\in\{10,70\}$) and a large number of chains ($N\in\{100,1000\}$), the MAE of IPCs approaches the MAE of O-MCMC. A possible explanation is that the interaction is particularly useful with small $N$ and a wrong choice of $\sigma$, whereas the use of large number of chains such as $N=100$ or $N=1000$ is enough, in this bidemensional example, for obtaining good performance. However, note that O-MCMC provides the lower MAE, is any cases.

| | **O-MCMC with SMH** | | | | | | **Independent parallel chains (IPCs)** | | |
|---|---|---|---|---|---|---|---|---|---|
| $N$ | 5 | | 100 | | 1000 | | 5 | 100 | 1000 |
| $T_V$ | 1 | 100 | 1 | 100 | 1 | 100 | – | – | – |
| $\sigma=2$ | 0.9683 | 1.2301 | 1.1532 | 1.5253 | 2.3611 | 2.4586 | 4.1986 | 2.6931 | 2.6923 |
| $\sigma=5$ | 0.9612 | 1.1548 | 0.6658 | 0.7810 | 1.1442 | 1.1948 | 2.7590 | 1.3395 | 1.3367 |
| $\sigma=10$ | 0.8723 | 0.9435 | 0.2562 | 0.2652 | 0.0948 | 0.0941 | 1.1212 | 0.2759 | 0.0951 |
| $\sigma=70$ | 1.0731 | 1.1474 | 0.4832 | 0.4801 | 0.5078 | 0.5024 | 1.6394 | 0.6027 | 0.5432 |
| $T$ | 4000 | | | | | | 2400 | 2020 | 2002 |
| $E_T$ | $12\cdot10^3$ | | $20.2\cdot10^4$ | | $200.2\cdot10^4$ | | $12\cdot10^3$ | $20.2\cdot10^4$ | $200.2\cdot10^4$ |

**Table 8**. Mean Absolute Error (MAE) in the estimation of the mean of the target (first component), averaged over 1000 runs, using O-MCMC with SMH and IPCs, considering different values of $\sigma$ and $T_V$ (recall, we set $T_V=T_H$). The total number of evaluations of the target $E_T$ is the same for O-MCMC (where $E_T=\frac{T}{2}(N+1)$ since $T_V=T_H$) and IPCs (where $E_T=NT$).

### 7.2. Spectral Analysis

WORK IN PROGRESS (see next versions in vixra.org)

## 8. CONCLUSIONS

In this work, we have introduced a novel family of MCMC algorithms, named Orthogonal MCMC schemes, that incorporates "horizontal" MCMC transitions to share information among a cloud of parallel "vertical" MCMC chains. We have described different alternatives for exchanging information among independent parallel chains. Compared to the fully independent parallel chains approach, the novel interacting techniques show a more robust behavior with respect to the parameterization and better performance for different number of chains. One reason is that the novel algorithms provide a good trade-off between the use of an independent or random walk proposal density, i.e., between local an global explorations. We have considered two different approaches for the interaction among the chains: in the first one, a MCMC technique over the entire population is directly applied whereas, in the second one, the initial population $\mathcal{P}_t$ is used for building a suitable mixture density $\psi(\mathbf{x})$ employed as proposal function in the horizontal transitions. This second approach can be interpreted as an adaptive MCMC scheme where the location parameters of the $N$ components of mixture $\psi(\mathbf{x})$ are updated driven by $N$ parallel MCMC chains. The outputs

[5] We set $T_{train}=T_V$, i.e., the adaptation starts after that the samples of the first vertical period are collected. Thus, before of the first horizontal step, $\varphi_t(\mathbf{x})$ has been already updated.

[6] We use all the generated samples in the estimation without removing any "burn-in" period.

of these parallel chains are also employed in the approximation of the target. Furthermore, we have designed different parallel Multiple Try Metropolis (P-MTM) schemes using an independent proposal pdf, where the drawn candidates are "recycled" in order to reduce the overall computational cost. Finally, we have described two modified versions of O-MCMC for optimization and the application to big data problems. The ergodicity of all the proposed methodologies have been discussed and several numerical simulations have been provided showing the advantages of the novel approach.

## 9. ACKNOWLEDGEMENTS

This work has been supported by the ERC grant 239784 and AoF grant 251170, the Spanish government through the CONSOLIDER-INGENIO 2010 program (CSD2008-00010) and DISSECT project (TEC2012-38058-C03-01), the BBVA Foundation through the MG-FIAR project, obtained in "Primera Convocatoria de Ayudas Fundación BBVA a Investigadores, Innovadores y Creadores Culturale", and by the Grant 2014/23160-6 of São Paulo Research Foundation (FAPESP).

## 10. REFERENCES

[1] A. Doucet and X. Wang, "Monte Carlo methods for signal processing," *IEEE Signal Processing Magazine*, vol. 22, no. 6, pp. 152–170, Nov. 2005.

[2] J. Ye, A. Wallace, and J. Thompson, "Parallel Markov chain Monte Carlo computation for varying-dimension signal analysis," in *Proc. EUSIPCO 2009*, Glasgow (Scotland), 24–28 Aug. 2009, pp. 2673–2677.

[3] D. Luengo and L. Martino, "Fully adaptive Gaussian mixture Metropolis-Hastings algorithm," in *Proc. ICASSP 2013*, Vancouver (Canada), 2013, pp. 6148–6152.

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

[5] L. Martino, J. Read, and D. Luengo, "Independent doubly adaptive rejection Metropolis sampling within Gibbs sampling," *(to appear) IEEE Transactions on Signal Processing*, 2015.

[6] R. Craiu, J. Rosenthal, and C. Yang, "Learn from the neighbor: Parallel-chain and regional adaptive MCMC," *Journal of the American Statistical Association*, vol. 104, no. 448, pp. 1454–1466, 2009.

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

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

[9] P. Jacob, C. P. Robert, and M. H. Smith, "Using parallel computation to improve Independent Metropolis-Hastings based estimation," *Journal of Computational and Graphical Statistics*, vol. 3, no. 20, pp. 616–635, 2011.

[10] B. Calderhead, "A general construction for parallelizing Metropolis-Hastings algorithms," *Proceedings of the National Academy of Sciences of the United States of America (PNAS)*, vol. 111, no. 49, pp. 17408–17413, 2014.

[11] R. Bardenet, A. Doucet, and C. Holmes, "On Markov chain Monte Carlo methods for tall data," *arXiv:1505.02827*, 2015.

[12] W. Neiswanger, C. Wang, and E. Xing, "Asymptotically exact, embarrassingly parallel MCMC," *arXiv:1311.4780*, pp. 1–16, 21 Mar. 2014.

[13] X. Wang and D. B. Dunson, "Parallelizing MCMC via Weierstrass sampler," *arXiv:1312.4605v2*, 25 May 2014.

[14] X. Wang, F. Guo, K. A. Heller, and D. B. Dunson, "Parallelizing MCMC with random partition trees," *arXiv:1311.4780*, 2015.

[15] Steven L. Scott, Alexander W. Blocker, Fernando V. Bonassi, Hugh A. Chipman, Edward I. George, and Robert E. McCulloch, "Bayes and big data: The consensus Monte Carlo algorithm," in *EFaBBayes 250th conference*, 2013, vol. 16.

[16] L. Martino, V. Elvira, D. Luengo, A. Artes, and J. Corander, "Orthogonal MCMC algorithms," *IEEE Workshop on Statistical Signal Processing (SSP)*, pp. 364–367, June 2014.

[17] A. Jasra, D. A. Stephens, and C. C. Holmes, "On population-based simulation for static inference," *Statistics and Computing*, vol. 17, no. 3, pp. 263–279, 2007.

[18] J. Corander, M. Gyllenberg, and T. Koski, "Bayesian model learning based on a parallel MCMC strategy," *Statistics Computing*, vol. 16, pp. 355–362, 2006.

[19] J. Corander, M. Ekdahl, and T. Koski, "Parallel interacting mcmc for learning of topologies of graphical models," *Data Mining and Knowledge Discovery*, vol. 17, no. 3, pp. 431–456, 2008.

[20] J. S. Liu, F. Liang, and W. H. Wong, "The multiple-try method and local optimization in Metropolis sampling," *Journal of the American Statistical Association*, vol. 95, no. 449, pp. 121–134, March 2000.

[21] L. Martino, V. P. Del Olmo, and J. Read, "A multi-point metropolis scheme with generic weight functions," *Statistics and Probability Letters*, vol. 82, no. 7, pp. 1445–1453, 2012.

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

[23] S. K. Kirkpatrick, C. D. Gelatt Jr., and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, May 1983.

[24] M. Locatelli, "Convergence of a simulated annealing algorithm for continuous global optimization," *Journal of Global Optimization*, vol. 18, pp. 219–234, 2000.

[25] F. Liang, Y. Cheng, and G. Lin, "Simulated stochastic approximation annealing for global optimization with a square-root cooling schedule," *Journal of the American Statistical Association*, vol. 109, no. 506, pp. 847–863, 2014.

[26] N. Chopin, "A sequential particle filter for static models," *Biometrika*, vol. 89, pp. 539–552, 2002.

[27] P. Del Moral, Arnaud Doucet, and A. Jasra, "Sequential Monte Carlo samplers," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, no. 3, pp. 411–436, 2006.

[28] G. Roberts and R. Tweedie, "Exponential convergence of Langevin distributions and their discrete approximations," *Bernoulli*, vol. 2, no. 4, pp. 341–363, 1996.

[29] A. Mira, "On Metropolis-Hastings algorithms with delayed rejection," *Metron*, vol. LIX, no. 3-4, pp. 231–241, 2001.

[30] F. Liang, C. Liu, and R. Caroll, *Advanced Markov Chain Monte Carlo Methods: Learning from Past Samples*, Wiley Series in Computational Statistics, England, 2010.

[31] L. Martino, V. Elvira, D. Luengo, and J. Corander, "An adaptive population importance sampler: Learning from the uncertanity," *(to appear in) IEEE Transactions on Signal Processing; viXra.org:1405.0280*, 2015.

[32] G. R. Warnes, "The Normal Kernel Coupler: An adaptive Markov Chain Monte Carlo method for efficiently sampling from multi-modal distributions," *Technical Report*, 2001.

[33] R. Neal, "MCMC using ensembles of states for problems with fast and slow variables such as Gaussian process regression," *arXiv:1101.0387*, 2011.

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

[35] A. A. Barker, "Monte Carlo calculations of the radial distribution functions for a proton-electron plasma," *Australian Journal of Physics*, vol. 18, pp. 119–133, 1965.

[36] Víctor Elvira, Luca Martino, David Luengo, and Mónica F Bugallo, "Efficient multiple importance sampling estimators," *Signal Processing Letters, IEEE*, vol. 22, no. 10, pp. 1757–1761, 2015.

[37] Heikki Haario, Eero Saksman, and Johanna Tamminen, "An adaptive Metropolis algorithm," *Bernoulli*, vol. 7, no. 2, pp. 223–242, April 2001.

[38] O. Cappé, A. Guillin, J. M. Marin, and C. P. Robert, "Population Monte Carlo," *Journal of Computational and Graphical Statistics*, vol. 13, no. 4, pp. 907–929, 2004.

## A. STATIONARY DISTRIBUTION OF O-MCMC

Considering an extended state space $\mathbb{R}^{d_X \times N}$, we can interpret that O-MCMC yields a unique chain in $\mathbb{R}^{d_X \times N}$. Namely, one population of states at the $t$-th iteration represents one extended state of this unique chain. Here, we show that this chain, generated by O-MCMC, has the extended target density

$$\bar{\pi}_g(\mathbf{x}_1, \ldots, \mathbf{x}_N) \propto \prod_{n=1}^{N} \pi(\mathbf{x}_n),$$

as invariant pdf. First of all, for the sake of simplicity, we tackle a simpler case where two MCMC kernels $K_1(\mathbf{y}|\mathbf{x})$, $K_2(\mathbf{z}|\mathbf{y})$, with $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathcal{D} \in \mathbb{R}^{d_x}$, are used sequentially and where $\bar{\pi}(\cdot)$ is the invariant pdf. Namely, we consider of MCMC techniques which steps are summarized in the two conditional probabilities $K_1(\mathbf{y}|\mathbf{x})$ and $K_2(\mathbf{z}|\mathbf{y})$, such that

$$\int_{\mathcal{D}} K_1(\mathbf{y}|\mathbf{x})\bar{\pi}(\mathbf{x})d\mathbf{x} = \bar{\pi}(\mathbf{y}), \qquad \int_{\mathcal{D}} K_2(\mathbf{z}|\mathbf{y})\bar{\pi}(\mathbf{y})d\mathbf{y} = \bar{\pi}(\mathbf{z}).$$

We consider the sequential application of $K_1$ and $K_2$, i.e, first $\mathbf{y}' \sim K_1(\mathbf{y}|\mathbf{x})$ and then draw $\mathbf{z}' \sim K_2(\mathbf{z}|\mathbf{y}')$, i.e., the probability of transition from $\mathbf{z}$ to $\mathbf{x}$

$$T(\mathbf{z}|\mathbf{x}) = \int_{\mathcal{D}} K_2(\mathbf{z}|\mathbf{y})K_1(\mathbf{y}|\mathbf{x})d\mathbf{y}. \tag{17}$$

The target $\bar{\pi}$ is also invariant w.r.t. $T(\mathbf{z}|\mathbf{x})$. Indeed, we can write

$$\begin{aligned}
\int_{\mathcal{D}} T(\mathbf{z}|\mathbf{x})\bar{\pi}(\mathbf{x})d\mathbf{x} &= \int_{\mathcal{D}} \left[\int K_2(\mathbf{z}|\mathbf{y})K_1(\mathbf{y}|\mathbf{x})d\mathbf{y}\right] \bar{\pi}(\mathbf{x})d\mathbf{x}, \\
&= \int_{\mathcal{D}} K_2(\mathbf{z}|\mathbf{y}) \left[\int K_1(\mathbf{y}|\mathbf{x})\bar{\pi}(\mathbf{x})d\mathbf{x}\right] d\mathbf{y}, \\
&= \int_{\mathcal{D}} K_2(\mathbf{z}|\mathbf{y})\bar{\pi}(\mathbf{y})d\mathbf{y}, \\
&= \bar{\pi}(\mathbf{z}),
\end{aligned} \tag{18}$$

that is the definition of invariant pdf w.r.t. $T(\mathbf{z}|\mathbf{x})$. We can use the same arguments for the O-MCMC schemes, considering now a population of current states, i.e.,

$$\mathcal{P}_{t-1} = \{\mathbf{x}_{1,t-1}, \ldots, \mathbf{x}_{N,t-1}\}.$$

We denote the vertical MCMC kernels as $K_n(\mathbf{x}_{n,t}|\mathbf{x}_{n,t-1})$ with $\bar{\pi}$ invariant, whereas one horizontal kernel $K_H(\mathcal{P}_t|\mathcal{P}_{t-1})$ with invariant pdf the following extended target pdf

$$\bar{\pi}_g(\mathcal{P}) \propto \pi_g(\mathbf{x}_1, \ldots, \mathbf{x}_N) = \prod_{n=1}^{N} \pi(\mathbf{x}_n).$$

The complete kernel of the orthogonal procedure, formed by one vertical and one orthogonal step, is

$$T(\mathcal{P}_t|\mathcal{P}_{t-2}) = \int_{\mathcal{D}^N} K_H(\mathcal{P}_t|\mathcal{P}_{t-1}) \left[\prod_{n=1}^{N} K_n(\mathbf{x}_{n,t-1}|\mathbf{x}_{n,t-2})\right] \prod_{n=1}^{N} d\mathbf{x}_{n,t-1}.$$

In this case, we can write

$$\int_{\mathcal{D}^N} T(\mathcal{P}_t|\mathcal{P}_{t-2})\bar{\pi}_g(\mathcal{P}_{t-2})d\mathcal{P}_{t-2} =$$

$$= \int_{\mathcal{D}^N} \int_{\mathbb{R}^N} K_H(\mathcal{P}_t|\mathcal{P}_{t-1}) \prod_{n=1}^{N} K_n(\mathbf{x}_{n,t-1}|\mathbf{x}_{n,t-2}) \prod_{n=1}^{N} \bar{\pi}(\mathbf{x}_{n,t-2}) \prod_{n=1}^{N} d\mathbf{x}_{n,t-1} \prod_{n=1}^{N} d\mathbf{x}_{n,t-2}.$$

$$= \int_{\mathcal{D}^N} K_H(\mathcal{P}_t|\mathcal{P}_{t-1}) \prod_{n=1}^{N} \bar{\pi}(\mathbf{x}_{n,t-1}) \prod_{n=1}^{N} d\mathbf{x}_{n,t-1} = \int_{\mathbb{R}^N} K_H(\mathcal{P}_t|\mathcal{P}_{t-1})\bar{\pi}_g(\mathcal{P}_{t-1})d\mathcal{P}_{t-1} = \bar{\pi}_g(\mathcal{P}_t).$$

Namely, the kernel $T(\mathcal{P}_t|\mathcal{P}_{t-2})$ has $\bar{\pi}_g$ as invariant density. This result can be easily extended when $T_V$ vertical and $T_H$ horizontal transitions are applied, using the same arguments.

## B. DISTRIBUTION AFTER RESAMPLING

Resampling procedures are employed in different Monte Carlo techniques such as Population Monte Carlo (PMC), Iterated Batch Importance Sampler (IBIS) and, more generally, in Sequential Monte Carlo (SMC) methods for a static scenario [38, 26, 27]. For simplicity, let us consider here a standard PMC-type scheme. In PMC, $N$ different proposal pdfs $q_1, \ldots, q_N$ are employed at each iteration. Starting from $\{\mathbf{x}_{1,0}, \ldots, \mathbf{x}_{N,0}\}$, the basic PMC scheme consists of the following steps:

1. For $t = 1, \ldots, T$:
   (a) For $n = 1, \ldots, N$, draw one sample $\mathbf{x}_{n,t}$ from $q_n$, i.e.,
   $$\mathbf{x}_{n,t} \sim q_n(\mathbf{x}|\mathbf{x}_{n,t-1}),$$
   (b) Draw $N$ independent samples $\{\mathbf{z}_1, \ldots, \mathbf{z}_N\}$ such that each $\mathbf{z}_n \in \{\mathbf{x}_{1,t}, \ldots, \mathbf{z}_{N,t}\}$, with $n = 1, \ldots, N$, with probability
   $$\beta_n = \frac{\frac{\pi(\mathbf{x}_{n,t})}{q_n(\mathbf{x}_{n,t})}}{\sum_{n=1}^{N} \frac{\pi(\mathbf{x}_{n,t})}{q_n(\mathbf{x}_{n,t})}}. \tag{19}$$
   (c) Set $\mathbf{x}_{n,t} = \mathbf{z}_n$.

The step 2(b) corresponds to resample (with replacement) $N$ times the population $\{\mathbf{x}_{n,t}\}_{n=1}^{N}$. Note that the weights in Eq. (19) are the same used in (10). For the sake of simplicity, since here we consider a generic iteration $t$, let us simplify the notation denoting as $\mathbf{x}_n = \mathbf{x}_{n,t} \sim q_n(\mathbf{x}|\mathbf{x}_{n,t-1})$ ($1 \leq n \leq N$, $1 \leq t \leq T$), and as $q_n(\mathbf{x}) = q_n(\mathbf{x}|\mathbf{x}_{n,t-1})$. Moreover, we define as

$$\mathbf{m}_{\neg n} = [\mathbf{x}_1, \ldots, \mathbf{x}_{n-1}, \mathbf{x}_{n+1}, \ldots, \mathbf{x}_N],$$

the matrix containing all the samples except for the $n$-th. Let us also denote as $\mathbf{z} \in \{\mathbf{x}_1 \ldots, \mathbf{x}_N\}$, a generic sample after applying one multinomial resampling step. Hence, the distribution of $\mathbf{z}$ is given by

$$\phi(\mathbf{z}) = \int_{\mathcal{D}^N} \hat{\pi}^{(N)}(\mathbf{z}) \left[\prod_{n=1}^{N} q_n(\mathbf{x}_n)\right] d\mathbf{x}_1 \ldots d\mathbf{x}_N, \tag{20}$$

where

$$\hat{\pi}^{(N)}(\mathbf{z}) = \sum_{j=1}^{N} \beta_j \delta(\mathbf{z} - \mathbf{x}_j), \tag{21}$$

and $\beta_j$ are given in Eq. (19). Then, after some straightforward rearrangements, Eq. (22) can be rewritten as

$$\phi(\mathbf{z}) = \sum_{j=1}^{N} \left( \int_{\mathcal{D}^{N-1}} \frac{\frac{\pi(\mathbf{x}_j)}{q_j(\mathbf{x}_j)}}{\sum_{n=1}^{N} \frac{\pi(\mathbf{x}_n)}{q_n(\mathbf{x}_n)}} \left[\prod_{\substack{n=1 \\ n \neq j}}^{N} q_n(\mathbf{x}_n)\right] d\mathbf{m}_{\neg j} \right) \delta(\mathbf{z} - \mathbf{x}_j). \tag{22}$$

Finally, we can write

$$\phi(\mathbf{z}) = \pi(\mathbf{z}) \sum_{j=1}^{N} \int_{\mathcal{D}^{N-1}} \frac{1}{N\hat{Z}} \left[\prod_{\substack{n=1 \\ n \neq j}}^{N} q_n(\mathbf{x}_n)\right] d\mathbf{m}_{\neg j}, \tag{23}$$

where $\hat{Z} = \frac{1}{N} \sum_{n=1}^{N} \frac{\pi(\mathbf{x}_n)}{q_n(\mathbf{x}_n)}$ is the estimate of the normalizing constant of the target obtained by using the importance sampling technique. When $N \to \infty$, then $\hat{Z} \to Z$ [4], and thus $\phi(\mathbf{z}) \to \frac{1}{Z}\pi(\mathbf{z}) = \bar{\pi}(\mathbf{z})$. Clearly, there exists a discrepancy between $\phi(\mathbf{z})$ and $\bar{\pi}(\mathbf{z})$.

## C. ERGODICITY OF THE PARALLEL SCHEMES BASED ON MULTIPLE CANDIDATES

Similarly as in PMC, the parallel Ensemble MCMC (P-EnM) and Multiple Try Metropolis (P-MTM) schemes in Tables 4-5 are based on the particle approximations of the measure of the target. In both cases, $L$ independent samples $\mathbf{z}_1, \ldots, \mathbf{z}_L$ drawn from $\psi(\mathbf{x})$, i.e.,

$$\mathbf{z}_\ell \sim \psi(\mathbf{x}), \tag{24}$$

for $\ell = 1, \ldots, L$. Below, we show that P-EnM and P-MTM yield reversible chains with stationary density the generalized pdf $\bar{\pi}_g$, proving the detailed balance condition is satisfied [4].

### C.1. Parellel Multiple Try Metropolis

In P-MTM, we can define the particle approximation based on the set $\{\mathbf{z}_1, \ldots, \mathbf{z}_L\}$, i.e.,

$$\hat{\pi}^{(L)}(\mathbf{z}) = \sum_{\ell=1}^{L} \beta_\ell \delta(\mathbf{z} - \mathbf{z}_\ell), \tag{25}$$

where the normalized weights $\beta_\ell$'s are given in Eq (10). Note that, the expression above coincides with Eq. (21). Let us also denote as the matrix

$$\mathbf{m}_{\neg k} = [\mathbf{z}_1, \ldots, \mathbf{z}_{k-1}, \mathbf{z}_{k+1}, \ldots, \mathbf{z}_L],$$

containing all the samples $\mathbf{z}_\ell$'s with the exception of $\mathbf{z}_k$. The kernel of the horizontal parallel MTM scheme can be written as

$$K_H(\mathcal{P}_t|\mathcal{P}_{t-1}) = \prod_{n=1}^{N} K_n(\mathbf{x}_{n,t}|\mathbf{x}_{n,t-1}), \tag{26}$$

where $K_n(\mathbf{x}_{n,t}|\mathbf{x}_{n,t-1})$ is the MTM kernel of $n$-th chain. Namely, $K_n(\mathbf{z}|\mathbf{x})$ is the probability of the $n$-th chain of jumping from the state $\mathbf{x} = \mathbf{x}_{t-1}$ to $\mathbf{z} = \mathbf{z}_k \in \{\mathbf{z}_1, \ldots, \mathbf{z}_L\}$ (for simplicity, we consider here only the case $\mathbf{z} \neq \mathbf{x}$). Note that Eq. (26) holds since all the $\mathbf{z}_\ell$'s are both drawn and resampled independently (see steps 2(a) and 2(b) in Table 5). Thus, the conditional probability $K_n(\mathbf{z}|\mathbf{x})$ can be expressed as

$$\begin{aligned} K_n(\mathbf{z} = \mathbf{z}_k|\mathbf{x}) &= \sum_{\ell=1}^{L} K_n(\mathbf{z}_k|\mathbf{x}, k = \ell), \\ &= L \int_{\mathcal{D}^{L-1}} \left[\prod_{\ell=1}^{L} \psi(\mathbf{z}_\ell)\right] \hat{\pi}^{(L)}_{MTM}(\mathbf{z}_k)\ \alpha_n(\mathbf{x}, \mathbf{z}_k|\mathbf{m}_{\neg k})\ d\mathbf{m}_{\neg k}, \quad \text{for} \quad \mathbf{z} \neq \mathbf{x}. \end{aligned} \tag{27}$$

where the function $\alpha_n$ is given in Eq. (11) and we have considered the case $\mathbf{x}$ and $\mathbf{z}$ (the case, $\mathbf{z} = \mathbf{x}$ is straightforward). The factor $L$ is due of the exchangeability among the $L$ random candidates. Thus, we can also write

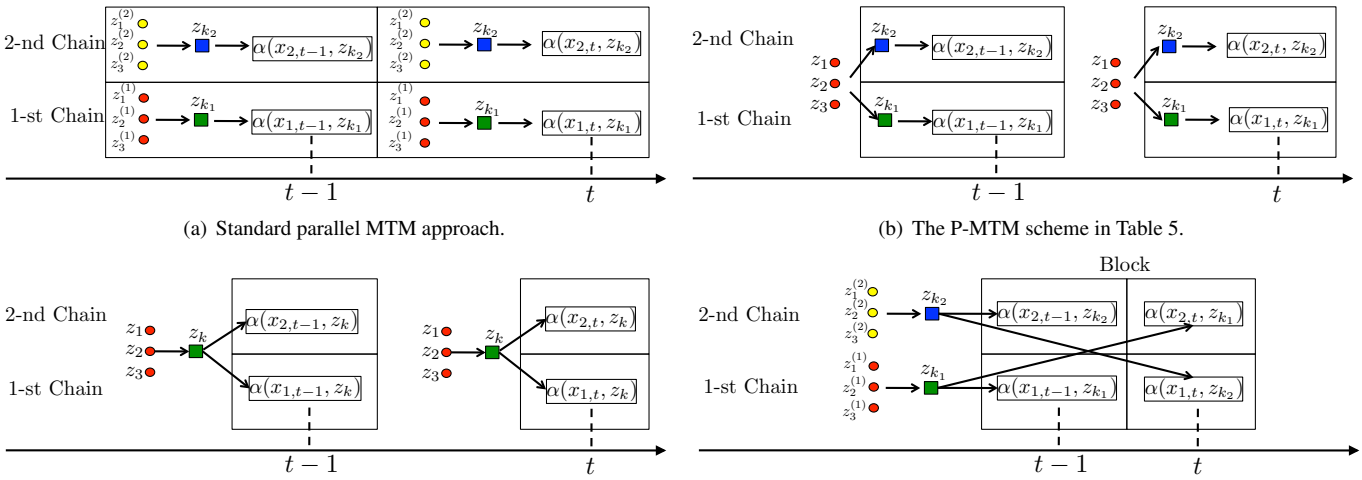
$$\begin{aligned} \bar{\pi}(\mathbf{x})K_n(\mathbf{z}_k|\mathbf{x}) &= L\bar{\pi}(\mathbf{x})\psi(\mathbf{z}_k) \int_{\mathcal{D}^{L-1}} \left[\prod_{\ell=1;\ell\neq k}^{L} \psi(\mathbf{z}_\ell)\right] \beta_k\ \alpha_n(\mathbf{x}, \mathbf{z}_k|\mathbf{m}_{\neg k})\ d\mathbf{m}_{\neg k}, \\ &= L\bar{\pi}(\mathbf{x})\psi(\mathbf{z}_k) \int_{\mathcal{D}^{L-1}} \left[\prod_{\ell=1;\ell\neq k}^{L} \psi(\mathbf{z}_\ell)\right] \frac{\frac{\pi(\mathbf{z}_k)}{\psi(\mathbf{z}_k)}}{\sum_{\ell=1}^{L} \frac{\pi(\mathbf{z}_\ell)}{\psi(\mathbf{z}_\ell)}}\ \alpha_n(\mathbf{x}, \mathbf{z}_k|\mathbf{m}_{\neg k})\ d\mathbf{m}_{\neg k}, \\ &= \frac{L}{Z}\pi(\mathbf{x})\pi(\mathbf{z}_k) \int_{\mathcal{D}^{L-1}} \left[\prod_{\ell=1;\ell\neq k}^{L} \psi(\mathbf{z}_\ell)\right] \frac{1}{\sum_{\ell=1}^{L} \frac{\pi(\mathbf{z}_\ell)}{\psi(\mathbf{z}_\ell)}}\ \alpha_n(\mathbf{x}, \mathbf{z}_k|\mathbf{m}_{\neg k})\ d\mathbf{m}_{\neg k}, \end{aligned} \tag{28}$$

where we have also used the equality $\bar{\pi}(\mathbf{x}) = \frac{1}{Z}\pi(\mathbf{x})$. Replacing

$$\alpha_n(\mathbf{x}, \mathbf{z}_k|\mathbf{m}_{\neg k}) = \min\left[1, \frac{\sum_{\ell=1}^{L} \frac{\pi(\mathbf{z}_\ell)}{\psi(\mathbf{z}_\ell)}}{\sum_{\ell=1}^{L} \frac{\pi(\mathbf{z}_\ell)}{\psi(\mathbf{z}_\ell)} - \frac{\pi(\mathbf{z}_k)}{\psi(\mathbf{z}_k)} + \frac{\pi(\mathbf{x})}{\psi(\mathbf{x})}}\right],$$

in the expression (28) and with some simple rearrangements, we obtain

$$\begin{aligned} \bar{\pi}(\mathbf{x})K_n(\mathbf{z}_k|\mathbf{x}) &= \frac{L}{Z}\pi(\mathbf{x})\pi(\mathbf{z}_k) \int_{\mathcal{D}^{L-1}} \left[\prod_{\ell=1;\ell\neq k}^{L} \psi(\mathbf{z}_\ell)\right] \\ &\quad \min\left[\frac{1}{\sum_{\ell\neq k}^{L} \frac{\pi(\mathbf{z}_\ell)}{\psi(\mathbf{z}_\ell)} + \frac{\pi(\mathbf{z}_k)}{\psi(\mathbf{z}_k)}}, \frac{1}{\sum_{\ell\neq k}^{L} \frac{\pi(\mathbf{z}_\ell)}{\psi(\mathbf{z}_\ell)} + \frac{\pi(\mathbf{x})}{\psi(\mathbf{x})}}\right]\ d\mathbf{m}_{\neg k}. \end{aligned} \tag{29}$$




(a) Standard parallel MTM approach.

(b) The P-MTM scheme in Table 5.

(c) Approach equivalent to BI-MTM scheme.

(d) The BI-MTM scheme in Table 6.

**Fig. 5**. A graphical representation of the several parallel MTM schemes with $N = 2$ chains and $L = 3$ tries. The BI-MTM scheme in **(d)** requires only 6 evaluations of the target pdf and 2 multinomial sampling steps considering two iterations, $t-1$ and $t$.

We can observe that, in equation above, we can exchange the position of the variables $\mathbf{x}$ so that $\mathbf{z}_k$ and the expression does not change. So that we can write

$$\bar{\pi}(\mathbf{x})K_n(\mathbf{z}_k|\mathbf{x}) = \bar{\pi}(\mathbf{z}_k)K_n(\mathbf{x}|\mathbf{z}_k),$$

for all $n = 1, \ldots, N$. As a consequence, we can also write

$$\bar{\pi}_g(\mathcal{P}_t)K_H(\mathcal{P}_t|\mathcal{P}_{t-1}) = \bar{\pi}_g(\mathcal{P}_{t-1})K_H(\mathcal{P}_{t-1}|\mathcal{P}_t), \tag{30}$$

where $K_H$ is defined in Eq. (26) and $\bar{\pi}_g$ is given in Eq. (5). The expression above is the so-called *detailed balance condition* [4]: since it holds, the horizontal MTM process has $\bar{\pi}_g$ as invariant pdf.

### *C.1.1. Important observations and Block Independent MTM*

First of all, note that with respect to a standard parallel multiple try approach, the novel P-MTM scheme generates only $L$ candidates at each iteration, instead of $NL$ samples. Indeed, P-MTM "recycles" the samples $\mathbf{z}_1, \ldots, \mathbf{z}_L$ from the independent proposal pdf $\psi(\mathbf{x})$, using them in all the $N$ chains. Namely, in P-MTM, at one iteration, the different MTM chains share the same set of tries. However, looking a single chain, each time $L$ new samples are drawn from $\psi(\mathbf{x})$ so that the chain is driven exactly from a standard (valid) MTM kernel. Figures 5(a) and (b) compare graphically the standard parallel MTM approach and te P-MTM scheme (with $N = 2$ chains and $L = 3$ tries). Observe that, in Figure 5(a), 12 new evaluations of the target are needed whereas only 6, in Figure 5(b).

Using the same arguments, the method remains valid if only one resampling step is performed at each iteration, providing one $\mathbf{z}^*$: in this case the same $\mathbf{z}^*$ is tested in the different acceptance tests of the $N$ parallel MTM chains, at the same iteration (exactly as in Table 2 and Fig. 3 for MH kernels). Figure 5(c) shows this case. In order to reduce the possible loss of the diversity, since several chains could jump at the same new state $\mathbf{z}^*$, an alternative strategy can be employed: the Block Independent MTM (BI-MTM) algorithm described in Table 6. Since the proposal $\psi$ is independent and then fixed, before a block of $N$ transitions, we can draw $NL$ tries from $\psi(\mathbf{x})$. Then, we can divide them in $N$ sets $\mathcal{S}_j$, with $j = 1, \ldots, N$ and select one sample from each set, obtaining $\{\mathbf{z}_{k_1}, \ldots, \mathbf{z}_{k_N}\}$ with $\mathbf{z}_{k_j} \in \mathcal{S}_j$. Then, we use $N$ different permutations of $\{\mathbf{z}_{k_1}, \ldots, \mathbf{z}_{k_N}\}$ for performing $N$ iterations of the $N$ parallel chains, providing a better mixing with respect to the case in Figure 5(c). This strategy, i.e., the BI-MTM scheme, is perfectly equivalent to the previous one, shown in Figure 5(c), from a theoretical and computational point of view. BI-MTM is represented graphically in Figure 5(d).

### C.2. Parallel Ensemble MCMC

Let us consider now the method in Table 4. In this case, the particle approximation is

$$\begin{aligned}\hat{\pi}_n^{(L+1)}(\mathbf{z}) &= \sum_{\ell=1}^{L} \alpha_\ell \delta(\mathbf{z}-\mathbf{z}_\ell) + \alpha_{L+1}\delta(\mathbf{z}-\mathbf{x}_{n,t-1}) \\ &= \sum_{\ell=1}^{L+1} \alpha_\ell \delta(\mathbf{z}-\mathbf{z}_\ell), \quad \text{where} \quad \mathbf{z}_{L+1}=\mathbf{x}_{n,t-1}, \end{aligned} \tag{31}$$

where the normalized weights $\alpha_\ell$'s are given in Eqs. (8)-(9). Again, the kernel of the P-EnM scheme can be written as

$$K_H(\mathcal{P}_t|\mathcal{P}_{t-1}) = \prod_{n=1}^{N} K_n(\mathbf{x}_{n,t}|\mathbf{x}_{n,t-1}),$$

where $K_n(\mathbf{x}_{n,t}|\mathbf{x}_{n,t-1})$ is the EnM kernel of $n$-th chain. In this case, for a given $n=1,\dots,N$, the conditional probability $K_n(\mathbf{z}=\mathbf{z}_k|\mathbf{x})$, where $\mathbf{x}=\mathbf{x}_{n,t-1}$ and $\mathbf{z}_k \in \{\mathbf{z}_1,\dots,\mathbf{z}_L,\mathbf{z}_{L+1}=\mathbf{x}_{n,t-1}\}$, is given by

$$\begin{aligned} K_n(\mathbf{z}_k|\mathbf{x}) &= \sum_{\ell=1}^{L} K_n(\mathbf{z}_k|\mathbf{x},k=\ell), \\ &= L\int_{\mathcal{D}^{L-1}} \left[\prod_{\ell=1}^{L}\psi(\mathbf{z}_\ell)\right] \hat{\pi}_n^{(L+1)}(\mathbf{z}_k)\; d\mathbf{m}_{\neg k}, \quad \text{for} \quad \mathbf{z}\neq\mathbf{x}. \end{aligned} \tag{32}$$

After some simple rearrangements (similarly in P-MTM) and using the formula of the weights in Eq. (8), we obtain

$$\begin{aligned} \bar{\pi}(\mathbf{x})K_n(\mathbf{z}_k|\mathbf{x}) &= L\bar{\pi}(\mathbf{x})\psi(\mathbf{z}_k)\int_{\mathcal{D}^{L-1}} \left[\prod_{\ell=1,\ell\neq k}^{L}\psi(\mathbf{z}_\ell)\right] \frac{\frac{\pi(\mathbf{z}_k)}{\psi(\mathbf{z}_k)}}{\sum_{\ell=1}^{L}\frac{\pi(\mathbf{z}_\ell)}{\psi(\mathbf{z}_\ell)}+\frac{\pi(\mathbf{x})}{\psi(\mathbf{x})}}\; d\mathbf{m}_{\neg k}, \\ &= \frac{L}{Z}\pi(\mathbf{x}\pi(\mathbf{z}_k)\int_{\mathcal{D}^{L-1}} \left[\prod_{\ell=1,\ell\neq k}^{L}\psi(\mathbf{z}_\ell)\right] \frac{1}{\sum_{\ell=1;\ell\neq\mathbf{z}_k}^{L}\frac{\pi(\mathbf{z}_\ell)}{\psi(\mathbf{z}_\ell)}+\frac{\pi(\mathbf{z}_k)}{\psi(\mathbf{z}_k)}+\frac{\pi(\mathbf{x})}{\psi(\mathbf{x})}}\; d\mathbf{m}_{\neg k}. \end{aligned} \tag{33}$$

Observing the last equation, we can clearly replace the variable $\mathbf{x}$ with $\mathbf{z}_k$ and vice versa, without changing the expression. Hence, finally we obtain

$$\bar{\pi}(\mathbf{x})K_n(\mathbf{z}_k|\mathbf{x}) = \bar{\pi}(\mathbf{z}_k)K_n(\mathbf{x}|\mathbf{z}_k),$$

for all $n=1,\dots,N$ and, as a consequence,

$$\bar{\pi}_g(\mathcal{P}_t)K_H(\mathcal{P}_t|\mathcal{P}_{t-1}) = \bar{\pi}_g(\mathcal{P}_{t-1})K_H(\mathcal{P}_{t-1}|\mathcal{P}_t), \tag{34}$$

that is the detailed balance condition. For further considerations, see App. C.1.1 above.