

# Non-reversible jump algorithms for Bayesian nested model selection

Philippe Gagnon

and

Arnaud Doucet

Department of Statistics, University of Oxford, United Kingdom

August 12, 2020

## Abstract

Non-reversible Markov chain Monte Carlo methods often outperform their reversible counterparts in terms of asymptotic variance of ergodic averages and mixing properties. Lifting the state-space (Chen et al., 1999; Diaconis et al., 2000) is a generic technique for constructing such samplers. The idea is to think of the random variables we want to generate as position variables and to associate to them direction variables so as to design Markov chains which do not have the diffusive behaviour often exhibited by reversible schemes. In this paper, we explore the benefits of using such ideas in the context of Bayesian model choice for nested models, a class of models for which the model indicator variable is an ordinal random variable. By lifting this model indicator variable, we obtain *non-reversible jump algorithms*, a non-reversible version of the popular reversible jump algorithms introduced by Green (1995). This simple algorithmic modification provides samplers which can empirically outperform their reversible counterparts at no extra computational cost. The code to reproduce all experiments is available online.<sup>1</sup>

*Keywords:* Bayesian statistics; Markov chain Monte Carlo methods; non-reversible Markov chains; Peskun–Tierney ordering; weak convergence.

---

<sup>1</sup>See ancillary files on [arXiv:1911.01340](https://arxiv.org/abs/1911.01340).

# 1 Introduction

Reversible jump (RJ) algorithms are a popular class of Markov chain Monte Carlo (MCMC) methods introduced by [Green \(1995, 2003\)](#). They are used to sample from a target distribution  $\pi$  defined on  $\bigcup_{j \in \mathcal{K}} \{j\} \times \mathbb{R}^{d_j}$ ,  $\mathcal{K}$  being a countable set. In the statistics applications discussed in this paper, this distribution corresponds to the joint posterior distribution of a model indicator  $k \in \mathcal{K}$  and its corresponding parameters  $\mathbf{x}_k \in \mathbb{R}^{d_k}$ . These samplers thus allow us to perform simultaneously model selection and parameter estimation. In the following, we assume for simplicity that the parameters of all models are continuous random variables and abuse notation by also using  $\pi$  to denote the target density.

Given the current state  $(k, \mathbf{x}_k)$ , a RJ algorithm generates the next state by proposing a model candidate  $k'$  from some probability mass function (PMF)  $g(k, \cdot)$  then a proposal for its corresponding parameter values. This last step is usually achieved through two sub-steps:

1. generate  $\mathbf{u}_{k \mapsto k'} \sim q_{k \mapsto k'}$  (this vector corresponds to auxiliary variables used, for instance, to propose values for additional parameters when  $d_{k'} > d_k$ ), where  $q_{k \mapsto k'}$  is a probability density function (PDF),
2. apply the function  $T_{k \mapsto k'}$  to  $(\mathbf{x}_k, \mathbf{u}_{k \mapsto k'})$ ,  $T_{k \mapsto k'}(\mathbf{x}_k, \mathbf{u}_{k \mapsto k'}) =: (\mathbf{y}_{k'}, \mathbf{u}_{k' \mapsto k})$ , where the vector  $\mathbf{y}_{k'}$  represents the proposal for the parameters of model  $k'$  and  $T_{k \mapsto k'}$  is a diffeomorphism (i.e. a differentiable map having a differentiable inverse).

The notation  $k \mapsto k'$  in subscript is used to highlight a dependance on both the current and proposed models. When  $k' = k$ , we say that a *parameter update* is proposed, whereas we say that a *model switch* is proposed when  $k' \neq k$ . The proposal  $(k', \mathbf{y}_{k'})$  is accepted with probability:

$$\alpha_{\text{RJ}}((k, \mathbf{x}_k), (k', \mathbf{y}_{k'})) := 1 \wedge \frac{g(k', k) \pi(k', \mathbf{y}_{k'}) q_{k' \mapsto k}(\mathbf{u}_{k' \mapsto k})}{g(k, k') \pi(k, \mathbf{x}_k) q_{k \mapsto k'}(\mathbf{u}_{k \mapsto k'}) |J_{T_{k \mapsto k'}}(\mathbf{x}_k, \mathbf{u}_{k \mapsto k'})|^{-1}}, \quad (1)$$

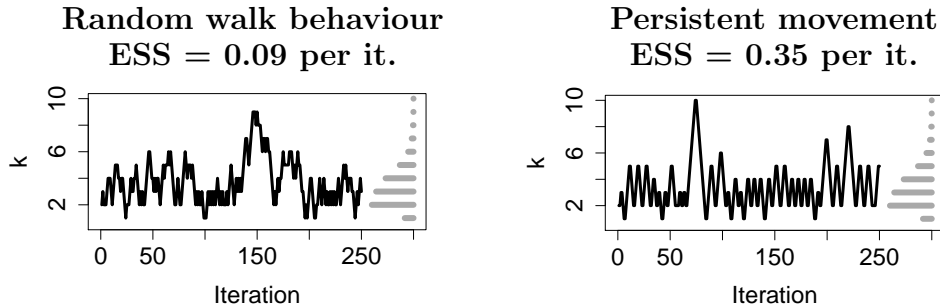
where  $x \wedge y := \min(x, y)$  and  $|J_{T_{k \mapsto k'}}(\mathbf{x}_k, \mathbf{u}_{k \mapsto k'})|$  is the absolute value of the determinant of the Jacobian matrix of the function  $T_{k \mapsto k'}$ . If the proposal is rejected, the chain remains at the same state  $(k, \mathbf{x}_k)$ .

In this paper, we consider the special case of nested models; i.e.  $K$  is an ordinal discrete random variable that reflects the complexity of the models. For instance, it represents the number of change-points in multiple change-point problems (Section 4, [Green \(1995\)](#)), the number of components in mixture modelling ([Richardson and Green, 1997](#)), the order of an autoregressive process ([Vermaak et al., 2004](#)), the number of clusters in dependence structures for multivariate extremes ([Vettori et al., 2019](#)) or the number of principal components included in robust principal component regression ([Gagnon et al., 2020](#)). We restrict our attention to samplers that switch models by taking steps of  $\pm 1$ , i.e.  $k' \in \{k-1, k+1\}$  when a model switch is proposed. This is a common choice which implies that the model space  $\mathcal{K}$  is explored through a random walk, a process that often backtracks and thus exhibits a diffusive behaviour. This choice of neighbourhood  $k' \in \{k-1, k+1\}$  in RJ makes the process reversible with respect to  $\pi$  and thus ensures that  $\pi$  is an invariant distribution.

The objective of this paper is to propose sampling schemes which do not suffer from such a diffusive behaviour by exploiting the lifting idea introduced by [Chen et al. \(1999\)](#) and [Diaconis et al. \(2000\)](#) to induce persistent movement in the model indicator. In the somewhat related contexts of simulated tempering ([Sakai and Hukushima, 2016](#)) and parallel tempering ([Syed et al., 2019](#)), lifting the temperature variable provides non-reversible samplers which perform substantially better than their reversible counterparts.

The changes that we make to the RJ sampling framework described above to apply the lifting idea are remarkably simple and require no additional computational effort. First, we extend the state-space by adding a direction variable  $\nu \in \{-1, 1\}$  and assign it a uniform distribution  $\mathcal{U}\{-1, 1\}$ . Second, when a model switch is proposed and the current state is  $(k, \mathbf{x}_k, \nu)$ , the model to explore is selected deterministically instead of randomly by setting:  $k' := k + \nu$ . If the proposal for the model to explore next (model  $k'$ ) along with its parameter values  $\mathbf{y}_{k'}$  is accepted, the next state of the chain is  $(k', \mathbf{y}_{k'}, \nu)$ . The direction for the model indicator remains the same; this is what induces persistent movement. If the proposal is rejected, the next state of the chain is  $(k, \mathbf{x}_k, -\nu)$ , so the direction is reversed for  $K$ . A proposal may be rejected because there is negligible mass beyond  $k$  in the direction followed; a change in direction may thus imply a return towards the high probability area.

Such simple modifications lead to a non-reversible scheme and can be very efficient as illustrated in Figure 1 (in which ESS stands for effective sample size). ESS per iteration is defined as the inverse of the integrated autocorrelation time. In this paper, it is used to evaluate algorithms regarding how efficient they are at sampling  $K$ . In particular, it measures their capacity of making the stochastic process  $\{K(m) : m \in \mathbb{N}\}$  traverse the model space, which is what we want to highlight.



**Figure 1:** Trace plots for *ideal* RJ and NRJ (we define what we mean by *ideal* in Section 2.2), and showing only the iterations in which model switches are proposed; the horizontal lines represent the marginal targeted PMF  $\pi(k)$  in a real multiple change-point problem presented in Section 5.2

The rest of this paper is organised as follows. We first introduce in Section 2 a general non-reversible jump (NRJ) algorithm and establish its validity. We also present its *ideal* version that is able to propose model parameter values from the conditional distributions  $\pi(\cdot | k)$ . This ideal algorithm is simple and allows us to exploit existing theoretical results to establish in Section 4 that NRJ can outperform the corresponding ideal RJ under some assumptions on the marginal PMF  $\pi(k)$ . Although such an ideal sampler cannot be implemented in practice for complex models, we show in Section 3 how we can leverage methods that have been previously developed in the RJ literature to approximate this ideal NRJ sampler. The weak convergence of the resulting sampler towards the ideal NRJ sampler is established as a precision parameter increases without bounds. In Section 4, we also prove that any NRJ (ideal or non-ideal) performs at least as good as its reversible counterpart. We present in Section 5 numerical experiments to illustrate the performance of NRJ samplers on a toy example and a real multiple change-point problem. We provide a discussion of implementation aspects and possible extensions in Section 6. All proofs of theoretical

results are provided in Section 1 of the supplementary material.

## 2 Non-reversible jump algorithms and ideal samplers

### 2.1 Non-reversible jump schemes

[Algorithm 1](#) presents the general NRJ which takes as inputs an initial state  $(k, \mathbf{x}_k, \nu)$ , a total number of iterations, the functions  $q_{k \mapsto k'}$  and  $T_{k \mapsto k'}$ , and  $0 \leq \tau \leq 1$  which represents the probability of proposing a parameter update at any given iteration. In trans-dimensional samplers, the probability of proposing a parameter update is typically allowed to depend on the current state. For ease of presentation, it is considered constant here.

---

#### Algorithm 1 NRJ

---

1. Sample  $u_c \sim \mathcal{U}(0, 1)$ .
  - 2.(a) If  $u_c \leq \tau$ , attempt a parameter update using a MCMC kernel of invariant distribution  $\pi(\cdot \mid k)$  while keeping the values of the model indicator  $k$  and direction  $\nu$  fixed.
  - 2.(b) If  $u_c > \tau$ , attempt a model switch from model  $k$  to model  $k' = k + \nu$ . Sample  $\mathbf{u}_{k \mapsto k'} \sim q_{k \mapsto k'}$  and  $u_a \sim \mathcal{U}(0, 1)$ , and compute  $(\mathbf{y}_{k'}, \mathbf{u}_{k' \mapsto k}) = T_{k \mapsto k'}(\mathbf{x}_k, \mathbf{u}_{k \mapsto k'})$ . If
 
$$u_a \leq \alpha_{\text{NRJ}}((k, \mathbf{x}_k), (k', \mathbf{y}_{k'})) := 1 \wedge \frac{\pi(k', \mathbf{y}_{k'}) q_{k' \mapsto k}(\mathbf{u}_{k' \mapsto k})}{\pi(k, \mathbf{x}_k) q_{k \mapsto k'}(\mathbf{u}_{k \mapsto k'}) |J_{T_{k \mapsto k'}}(\mathbf{x}_k, \mathbf{u}_{k \mapsto k'})|^{-1}}, \quad (2)$$
 set the next state of the chain to  $(k', \mathbf{y}_{k'}, \nu)$ . Otherwise, set it to  $(k, \mathbf{x}_k, -\nu)$ .
  3. Go to Step 1.
- 

[Proposition 1](#) below ensures that [Algorithm 1](#) targets the correct distribution. Note that  $\mathbf{x}_k$  can be vectors containing both position and velocity variables, which allows using Hamiltonian Monte Carlo (HMC, see, e.g., [Neal \(2011\)](#)) and more generally discrete-time piecewise-deterministic MCMC schemes ([Vanetti et al., 2017](#)) for updating the parameters within [Algorithm 1](#). The only prerequisite is that the method leaves the conditional distributions  $\pi(\cdot \mid k)$  invariant. The proof of [Proposition 1](#) establishes that any valid scheme

used for parameter proposals during model switches in RJ framework, such as those of Karagiannis and Andrieu (2013) and Andrieu et al. (2018) presented in Section 3, are also valid in the non-reversible framework.

**Proposition 1** (Invariance). *The transition kernel of the Markov chain  $\{(K, \mathbf{X}_K, \nu)(m) : m \in \mathbb{N}\}$  simulated by Algorithm 1 admits  $\pi \otimes \mathcal{U}\{-1, 1\}$  as invariant distribution.*

Nothing prevents Algorithm 1 from switching to models at a distance of more than 1, i.e. with  $|k' - k| > 1$ . In Step 2.(b), an additional random variable  $\omega \in \{0, 1, \dots\}$  can be independently generated from, for instance, a Poisson distribution with a given mean parameter. In this case, we attempt to make a transition to model  $k' = k + \omega\nu$ , but nothing else changes and the algorithm is still valid. In practice, however,  $|d_{k'} - d_k|$  typically increases with  $|k' - k|$ , requiring to design proposal distributions of high dimensions, which is often very difficult and motivates using jumps to models no further than  $k \pm 1$  as in Green (1995), Richardson and Green (1997), Vermaak et al. (2004), Vettori et al. (2019) and Gagnon et al. (2020).

## 2.2 Ideal samplers and their advantages

When switching models, one would ideally be able to sample from the correct conditional distributions  $\pi(\cdot | k')$  to propose parameter values  $\mathbf{y}_{k'}$ . In this ideal situation, one can set  $q_{k \rightarrow k'} := \pi(\cdot | k')$ ,  $q_{k' \rightarrow k} := \pi(\cdot | k)$ , and  $T_{k \rightarrow k'}$  such that  $\mathbf{y}_{k'} := \mathbf{u}_{k \rightarrow k'}$  (which implies that  $\mathbf{u}_{k' \rightarrow k} := \mathbf{x}_k$ ), and observe that the acceptance probabilities reduce to

$$\alpha_{\text{NRJ}}((k, \mathbf{x}_k), (k', \mathbf{y}_{k'})) = 1 \wedge \frac{\pi(k')}{\pi(k)}. \quad (3)$$

These probabilities are independent of the current and proposed parameters values: a model proposal  $k'$  is accepted solely on the basis of the ratio of marginal posterior probabilities.

In general, the acceptance probabilities are as above whenever

$$\frac{\pi(\mathbf{y}_{k'} | k') q_{k' \rightarrow k}(\mathbf{u}_{k' \rightarrow k})}{\pi(\mathbf{x}_k | k) q_{k \rightarrow k'}(\mathbf{u}_{k \rightarrow k'}) |J_{T_{k \rightarrow k'}}(\mathbf{x}_k, \mathbf{u}_{k \rightarrow k'})|^{-1}} = 1, \quad (4)$$

for any switch from model  $k$  with parameter values  $\mathbf{x}_k$  to model  $k' \neq k$  with parameter values  $\mathbf{y}_{k'}$ , using the auxiliary variables  $\mathbf{u}_{k \rightarrow k'}$  and  $\mathbf{u}_{k' \rightarrow k}$ . In this more general setting, we

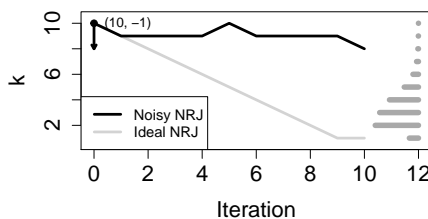
observe that (4) is verified if, starting with random variables distributed as  $\pi(\cdot | k) \otimes q_{k \mapsto k'}$  and applying the function  $T_{k \mapsto k'}$ , we obtain random variables distributed as  $\pi(\cdot | k') \otimes q_{k' \mapsto k}$ .

For realistic scenarios where the proposals  $\mathbf{y}_{k'}$  are not generated from the conditional distributions  $\pi(\cdot | k')$ , the acceptance probability of model switches for NRJ (2) can be expressed as a “noisy” version of that in (3):

$$\alpha_{\text{NRJ}}((k, \mathbf{x}_k), (k', \mathbf{y}_{k'})) = 1 \wedge \frac{\pi(k')}{\pi(k)} \varepsilon(k, k', \mathbf{x}_k, q_{k \mapsto k'}, q_{k' \mapsto k}, T_{k \mapsto k'}), \quad (5)$$

where  $\varepsilon$  represents multiplicative noise given by the left-hand side (LHS) in (4).

For NRJ to be beneficial, it will be useful to have a low variance noise. Imagine that the targeted marginal PMF is that on the right of Figure 2 and that the samplers are initialised at  $(K, \nu)(0) := (10, -1)$  (as in Figure 2), the advantage of the ideal NRJ is that it continues following the direction  $-1$  for several iterations as the ratios  $\pi(k-1)/\pi(k)$  are greater than 1 (which implies that the proposals are accepted). If the noise fluctuations are significant, such moves might be rejected.



**Figure 2:** Trace plots for “noisy” and ideal NRJ, and showing only the iterations in which model switches are proposed; the horizontal lines represent the marginal targeted PMF  $\pi(k)$  which is that in a real multiple change-point problem presented in Section 5.2

### 3 Towards ideal NRJ

We explained in the last section why it may be important to implement NRJ samplers that are close to their ideal counterparts, with low variance noise  $\varepsilon$ . We present in this section methods to achieve this by adapting some developed within the RJ framework. In Section 3.1, we present and adapt for NRJ the method of Karagiannis and Andrieu (2013). We proceed similarly in Section 3.2 with the approach of Andrieu et al. (2018). In

Section 3.3, we prove that, as  $\varepsilon \rightarrow 1$  in distribution, the Markov chains produced by NRJ incorporating these approaches converge weakly to the chains produced by ideal NRJ.

### 3.1 NRJ with the method of Karagiannis and Andrieu (2013)

Model  $k$  and model  $k'$  may be quite different. Jumping (“in one step”) from the former with parameters  $\mathbf{x}_k$  to the latter with parameters  $\mathbf{y}_{k'}$  may thus be difficult; i.e. starting with  $(\mathbf{x}_k, \mathbf{u}_{k \rightarrow k'}) \sim \pi(\cdot | k) \otimes q_{k \rightarrow k'}$ , it may be difficult to design a function  $T_{k \rightarrow k'}$  such that  $(\mathbf{y}_{k'}, \mathbf{u}_{k' \rightarrow k}) \sim \pi(\cdot | k') \otimes q_{k' \rightarrow k}$  approximately, for any reasonable choice of  $q_{k \rightarrow k'}$  and  $q_{k' \rightarrow k}$ .

Karagiannis and Andrieu (2013) introduce a sequence of auxiliary distributions playing the role of a specific class of imaginary models to ease transitions between model  $k$  and model  $k'$ . A proposal distribution is build by sampling an inhomogeneous Markov chain which targets at each step one of these auxiliary distributions in the spirit of annealed importance sampling (Neal, 2001). These auxiliary distributions take the form

$$\begin{aligned} \rho_{k \rightarrow k'}^{(t)}(\mathbf{x}_k^{(t)}, \mathbf{u}_{k \rightarrow k'}^{(t)}) &\propto \left[ \pi(k, \mathbf{x}_k^{(t)}) q_{k \rightarrow k'}(\mathbf{u}_{k \rightarrow k'}^{(t)}) |J_{T_{k \rightarrow k'}}(\mathbf{x}_k^{(t)}, \mathbf{u}_{k \rightarrow k'}^{(t)})|^{-1} \right]^{1-\gamma_t} \left[ \pi(k', \mathbf{y}_{k'}^{(t)}) q_{k' \rightarrow k}(\mathbf{u}_{k' \rightarrow k}^{(t)}) \right]^{\gamma_t}, \\ \rho_{k' \rightarrow k}^{(t)}(\mathbf{y}_{k'}^{(t)}, \mathbf{u}_{k' \rightarrow k}^{(t)}) &\propto \left[ \pi(k, \mathbf{x}_k^{(t)}) q_{k \rightarrow k'}(\mathbf{u}_{k \rightarrow k'}^{(t)}) |J_{T_{k \rightarrow k'}}(\mathbf{x}_k^{(t)}, \mathbf{u}_{k \rightarrow k'}^{(t)})|^{-1} \right]^{1-\gamma_{T-t}} \left[ \pi(k', \mathbf{y}_{k'}^{(t)}) q_{k' \rightarrow k}(\mathbf{u}_{k' \rightarrow k}^{(t)}) \right]^{\gamma_{T-t}}, \end{aligned} \quad (6)$$

for  $t = 0, \dots, T$  where  $T$  is a positive integer,  $\gamma_0 := 0, \gamma_T := 1$  and  $\gamma_t \in [0, 1]$  for  $t \in \{1, \dots, T-1\}$ . We set  $\gamma_t := t/T$  in our numerical experiments as in Karagiannis and Andrieu (2013). When switching from model  $k$  to model  $k'$ , we thus use at time  $t$  a transition kernel  $K_{k \rightarrow k'}^{(t)}$  to target the distribution  $\rho_{k \rightarrow k'}^{(t)}$ , which is at the beginning close to  $(\pi(k, \cdot) \otimes q_{k \rightarrow k'}) |J_{T_{k \rightarrow k'}}|^{-1}$ , and the end close to  $\pi(k', \cdot) \otimes q_{k' \rightarrow k}$ . We wrote  $\rho_{k \rightarrow k'}^{(t)}$  as a function of  $(\mathbf{x}_k^{(t)}, \mathbf{u}_{k \rightarrow k'}^{(t)})$  to emphasise that the starting point is  $(\mathbf{x}_k^{(0)}, \mathbf{u}_{k \rightarrow k'}^{(0)})$ . It is in fact also a function of  $(\mathbf{y}_{k'}^{(t)}, \mathbf{u}_{k' \rightarrow k}^{(t)})$  that can be found using  $(\mathbf{y}_{k'}^{(t)}, \mathbf{u}_{k' \rightarrow k}^{(t)}) = T_{k \rightarrow k'}(\mathbf{x}_k^{(t)}, \mathbf{u}_{k \rightarrow k'}^{(t)})$ .

The NRJ procedure incorporating such proposals is described in Algorithm 2. In Step 2.(b), the path can be generated through  $(\mathbf{y}_{k'}^{(t)}, \mathbf{u}_{k' \rightarrow k}^{(t)})$  instead.

Karagiannis and Andrieu (2013) explain that the MH correction term in Algorithm 2, that we denote by

$$r_{\text{NRJ2}}((k, \mathbf{x}_k^{(0)}), (k', \mathbf{y}_{k'}^{(T-1)})) := \prod_{t=0}^{T-1} \frac{\rho_{k \rightarrow k'}^{(t+1)}(\mathbf{x}_k^{(t)}, \mathbf{u}_{k \rightarrow k'}^{(t)})}{\rho_{k \rightarrow k'}^{(t)}(\mathbf{x}_k^{(t)}, \mathbf{u}_{k \rightarrow k'}^{(t)})}, \quad (7)$$

represents a consistent estimator of  $\pi(k')/\pi(k)$  as  $T \rightarrow \infty$ .



---

**Algorithm 2** NRJ incorporating the method of [Karagiannis and Andrieu \(2013\)](#)


---

1. Sample  $u_c \sim \mathcal{U}(0, 1)$ .
  - 2.(a) If  $u_c \leq \tau$ , attempt a parameter update using a MCMC kernel of invariant distribution  $\pi(\cdot \mid k)$  while keeping the values of the model indicator  $k$  and direction  $\nu$  fixed.
  - 2.(b) If  $u_c > \tau$ , attempt a model switch from model  $k$  to model  $k' := k + \nu$ . Sample  $\mathbf{u}_{k \rightarrow k'}^{(0)} \sim q_{k \rightarrow k'}$  and  $u_a \sim \mathcal{U}(0, 1)$ , and set  $\mathbf{x}_k^{(0)} := \mathbf{x}_k$ . Sample a path  $(\mathbf{x}_k^{(1)}, \mathbf{u}_{k \rightarrow k'}^{(1)}), \dots, (\mathbf{x}_k^{(T-1)}, \mathbf{u}_{k \rightarrow k'}^{(T-1)})$ , where  $(\mathbf{x}_k^{(t)}, \mathbf{u}_{k \rightarrow k'}^{(t)}) \sim K_{k \rightarrow k'}^{(t)}((\mathbf{x}_k^{(t-1)}, \mathbf{u}_{k \rightarrow k'}^{(t-1)}), \cdot)$ . Compute  $(\mathbf{y}_{k'}^{(t)}, \mathbf{u}_{k' \rightarrow k}^{(t)}) := T_{k \rightarrow k'}^{(t)}(\mathbf{x}_k^{(t)}, \mathbf{u}_{k \rightarrow k'}^{(t)})$  for  $t = 0, \dots, T-1$ . If  $u_a \leq \alpha_{\text{NRJ2}}((k, \mathbf{x}_k^{(0)}), (k', \mathbf{y}_{k'}^{(T-1)})) := 1 \wedge r_{\text{NRJ2}}((k, \mathbf{x}_k^{(0)}), (k', \mathbf{y}_{k'}^{(T-1)}))$  ( $r_{\text{NRJ2}}$  is defined in (7)), set the next state of the chain to  $(k', \mathbf{y}_{k'}^{(T-1)}, \nu)$ . Otherwise, set it to  $(k, \mathbf{x}_k, -\nu)$ .
  3. Go to Step 1.
- 

Under the following two conditions, the RJ corresponding to [Algorithm 2](#) and [Algorithm 2](#) itself are valid, in the sense that the target distribution is an invariant distribution. As mentioned in [Karagiannis and Andrieu \(2013\)](#), (8) below is verified if for all  $t$ ,  $K_{k \rightarrow k'}^{(t)}(\cdot, \cdot)$  and  $K_{k' \rightarrow k}^{(T-t)}(\cdot, \cdot)$  are Metropolis–Hastings (MH) kernels sharing the same proposal distributions.

**Symmetry condition:** For  $t = 1, \dots, T-1$  the pairs of transition kernels  $K_{k \rightarrow k'}^{(t)}(\cdot, \cdot)$  and  $K_{k' \rightarrow k}^{(T-t)}(\cdot, \cdot)$  satisfy

$$K_{k \rightarrow k'}^{(t)}((\mathbf{x}_k, \mathbf{u}_{k \rightarrow k'}), \cdot) = K_{k' \rightarrow k}^{(T-t)}((\mathbf{x}_k, \mathbf{u}_{k \rightarrow k'}), \cdot) \quad \text{for any } (\mathbf{x}_k, \mathbf{u}_{k \rightarrow k'}). \quad (8)$$

**Reversibility condition:** For  $t = 1, \dots, T-1$ , and for any  $(\mathbf{x}_k, \mathbf{u}_{k \rightarrow k'})$  and  $(\mathbf{x}'_k, \mathbf{u}'_{k \rightarrow k'})$ ,

$$\rho_{k \rightarrow k'}^{(t)}(\mathbf{x}_k, \mathbf{u}_{k \rightarrow k'}) K_{k \rightarrow k'}^{(t)}((\mathbf{x}_k, \mathbf{u}_{k \rightarrow k'}), (\mathbf{x}'_k, \mathbf{u}'_{k \rightarrow k'})) = \rho_{k' \rightarrow k}^{(t)}(\mathbf{x}'_k, \mathbf{u}'_{k \rightarrow k'}) K_{k' \rightarrow k}^{(t)}((\mathbf{x}'_k, \mathbf{u}'_{k \rightarrow k'}), (\mathbf{x}_k, \mathbf{u}_{k \rightarrow k'})). \quad (9)$$

The use of such sophisticated proposal schemes comes at a computational cost. As explained in [Karagiannis and Andrieu \(2013\)](#), the cost of using their approach is  $\mathcal{O}(I \times T)$ ,

$I$  denoting the number of iterations. Indeed, typically in Step 2.(b),  $T - 1$  MH steps similar to those used to update the parameters (Step 2.(a)) are applied. Fortunately, the improvement as a function of  $T$  for a fixed value of  $I$  may be very marked for  $T \leq T_0$ , leading to better results than one would obtain by instead setting  $T = 1$  (corresponding to [Algorithm 1](#) or vanilla RJ) and increasing  $I$  to attain the same computational budget. This is what is observed for the multiple change-point problem presented in [Section 5.2](#). The cost is indeed offset by a large enough improvement in terms of total variation between the empirical and true marginal model distributions for  $T$  in an interval including the value 100, which is the value used. That being said, the computational burden can be mitigated by designing better proposal functions  $q_{k \rightarrow k'}$  and  $T_{k \rightarrow k'}$  (when this is feasible), as shown in [Gagnon \(2019\)](#). In [Section 5.2](#), we only show the results of the sampler combining the approach of [Karagiannis and Andrieu \(2013\)](#) with that presented in the next section for brevity.

### 3.2 NRJ additionally with the method of [Andrieu et al. \(2018\)](#)

As mentioned,  $r_{\text{NRJ2}}$  in [\(7\)](#) can be interpreted as an estimator of  $\pi(k')/\pi(k)$ . To further reduce the variance of this estimator, one could produce in parallel  $N$  inhomogeneous Markov chains ending with  $N$  proposals, that we denote by  $\mathbf{y}_{k'}^{(T-1,1)}, \dots, \mathbf{y}_{k'}^{(T-1,N)}$ , and use instead the average of the  $N$  estimates  $r_{\text{NRJ2}}((k, \mathbf{x}_k^{(0)}), (k', \mathbf{y}_{k'}^{(T-1,1)})), \dots, r_{\text{NRJ2}}((k, \mathbf{x}_k^{(0)}), (k', \mathbf{y}_{k'}^{(T-1,N)}))$ . Simplifying notation, an estimate of  $\pi(k')/\pi(k)$  is thus given by

$$\bar{r}(k, k') := \frac{1}{N} \sum_{j=1}^N r_{\text{NRJ2}}((k, \mathbf{x}_k^{(0)}), (k', \mathbf{y}_{k'}^{(T-1,j)})).$$

However, applying this method naively does not lead to valid algorithms. The approach of [Andrieu et al. \(2018\)](#) exploits this averaging idea while leading to valid schemes. We present in [Algorithm 3](#) the NRJ version of this algorithm.

[Andrieu et al. \(2018\)](#) prove that increasing  $N$  decreases the asymptotic variance of the Monte Carlo estimates produced by RJ incorporating their approach. Their proof cannot be easily extended to NRJ. However, we have observed empirically that increasing  $N$  (as increasing  $T$  in [Algorithm 2](#)) leads to a steady increase in the ESS until the samplers are

---

**Algorithm 3** NRJ additionally incorporating the method of [Andrieu et al. \(2018\)](#)

---

1. Sample  $u_{c,1} \sim \mathcal{U}(0, 1)$ .
  - 2.(a) If  $u_{c,1} \leq \tau$ , attempt a parameter update using a MCMC kernel of invariant distribution  $\pi(\cdot \mid k)$  while keeping the values of the model indicator  $k$  and direction  $\nu$  fixed.
  - 2.(b) If  $u_{c,1} > \tau$ , attempt a model switch from model  $k$  to model  $k' := k + \nu$ . Sample  $u_a, u_{c,2} \sim \mathcal{U}(0, 1)$ . If  $u_{c,2} \leq 1/2$  go to Step 2.(b-i), otherwise go to Step 2.(b-ii).
  - 2.(b-i) Sample  $N$  proposals  $\mathbf{y}_{k'}^{(T-1,1)}, \dots, \mathbf{y}_{k'}^{(T-1,N)}$  as in Step 2.(b) of [Algorithm 2](#). Sample  $j^*$  from a PMF such that  $\mathbb{P}(J^* = j) \propto r_{\text{NRJ2}}((k, \mathbf{x}_k), (k', \mathbf{y}_{k'}^{(T-1,j)}))$ . If  $u_a \leq \bar{r}(k, k')$ , set the next state of the chain to  $(k', \mathbf{y}_{k'}^{(T-1,j^*)}, \nu)$ . Otherwise, set it to  $(k, \mathbf{x}_k, -\nu)$ .
  - 2.(b-ii) Sample one forward path as in Step 2.(b) of [Algorithm 2](#). Denote by  $\mathbf{y}_{k'}^{(T-1,1)}$  the endpoint. From  $\mathbf{y}_{k'}^{(T-1,1)}$ , generate  $N - 1$  reverse paths again as in Step 2.(b) of [Algorithm 2](#), yielding  $N - 1$  proposals for the parameters of model  $k$ . If  $u_a \leq \bar{r}(k', k)^{-1}$ , set the next state of the chain to  $(k', \mathbf{y}_{k'}^{(T-1,1)}, \nu)$ . Otherwise, set it to  $(k, \mathbf{x}_k, -\nu)$ .
  3. Go to Step 1.
- 

close enough to be ideal.

An advantage of the approach presented here over that presented in the previous section is that several computations can be executed in parallel. The  $N$  proposals in Step 2.(b-i) are indeed generated from the same starting point, implying that this part and the computations of the ratios  $r_{\text{NRJ2}}$  can be executed in parallel. Also, in Step 2.(b-ii), once  $\mathbf{y}_{k'}^{(T-1,1)}$  has been generated, the  $N - 1$  reverse paths can be generated in parallel. The computational cost associated to [Algorithm 3](#) is thus that of running [Algorithm 2](#) with the same value for  $T$  but with an additional 50% of model switches (because 50% of model switches use Step 2.(b-ii)), and to this we need to add the cost of computational overhead which depends on  $N$ . If [Algorithm 3](#) is run for  $I$  iterations, then its cost is upper bounded by that of running [Algorithm 2](#) for  $1.5I$  iterations plus the cost of computational overhead.

We for instance try implementing [Algorithm 3](#) using the R package `parallel` for the multiple change-point problem presented in [Section 5.2](#). This package provides an easy way of executing tasks in parallel. For this implementation, the computational cost is a little more than doubled.

### 3.3 Convergence of Algorithms 2 and 3 towards ideal NRJ

We presented at the beginning of [Section 3](#) intuitive reasons explaining why Algorithms 2 and 3 can be made as close as we want to their ideal counterparts. We present here theoretical arguments supporting this intuition by establishing the weak convergence of the Markov chains produced by [Algorithm 2](#) towards those simulated by its ideal version as  $T \rightarrow \infty$ . This implies that [Algorithm 3](#) with large enough  $T$  and fixed  $N$  generates Markov chains sharing the same behaviour as its ideal counterpart given that the noise  $\varepsilon$  (5) is only made more stable around the constant 1 by additionally using the approach of [Andrieu et al. \(2018\)](#). The corresponding weak convergence result for RJ incorporating the method of [Karagiannis and Andrieu \(2013\)](#) holds under the same assumptions as those presented here.

The Markov kernel simulated by [Algorithm 2](#) (when switching models) is given by:

$$\begin{aligned}
P_T((k, \mathbf{x}_k, \nu), (k', \mathbf{y}_{k'}, \nu')) &:= q_{k \mapsto k+\nu}(\mathbf{u}_{k \mapsto k+\nu}^{(0)}) \prod_{t=1}^{T-1} K_{k \mapsto k+\nu}^{(t)}((\mathbf{y}_{k+\nu}^{(t-1)}, \mathbf{u}_{k+\nu \mapsto k}^{(t-1)}), (\mathbf{y}_{k+\nu}^{(t)}, \mathbf{u}_{k+\nu \mapsto k}^{(t)})) \\
&\quad \times \delta_{(k+\nu, \mathbf{y}_{k+\nu}^{(T-1)}, \nu)}(k', \mathbf{y}_{k'}, \nu') \alpha_{\text{NRJ2}}((k, \mathbf{x}_k), (k', \mathbf{y}_{k'})) \\
&\quad + \delta_{(k, \mathbf{x}_k, -\nu)}(k', \mathbf{y}_{k'}, \nu') \int q_{k \mapsto k+\nu}(\mathbf{u}_{k \mapsto k+\nu}^{(0)}) \prod_{t=1}^{T-1} K_{k \mapsto k+\nu}^{(t)}((\mathbf{y}_{k+\nu}^{(t-1)}, \mathbf{u}_{k+\nu \mapsto k}^{(t-1)}), (\mathbf{y}_{k+\nu}^{(t)}, \mathbf{u}_{k+\nu \mapsto k}^{(t)})) \\
&\quad \times (1 - \alpha_{\text{NRJ2}}((k, \mathbf{x}_k), (k+\nu, \mathbf{y}_{k+\nu}^{(T-1)}))) d\mathbf{u}_{k \mapsto k+\nu}^{(0)} d(\mathbf{y}_{k+\nu}^{(1)}, \mathbf{u}_{k+\nu \mapsto k}^{(1)}) \dots d(\mathbf{y}_{k+\nu}^{(T-1)}, \mathbf{u}_{k+\nu \mapsto k}^{(T-1)}).
\end{aligned}$$

Use  $\{(K, \mathbf{X}_K, \nu)_T(m) : m \in \mathbb{N}\}$  to denote the Markov chain associated with this kernel. The ideal version of [Algorithm 2](#) presented in [Section 2.2](#) sets  $q_{k \mapsto k'} := \pi(\cdot \mid k')$ . This is when switching models. For the parameter update step, we assume that both samplers use the same MCMC kernels of invariant distributions  $\pi(\cdot \mid k)$ . The Markov kernel simulated by the ideal version (when switching models) is thus given by:

$$P_{\text{ideal}}((k, \mathbf{x}_k, \nu), (k', \mathbf{y}_{k'}, \nu')) := \pi(\mathbf{u}_{k \mapsto k+\nu} \mid k+\nu) \delta_{(k+\nu, \mathbf{u}_{k \mapsto k+\nu}, \nu)}(k', \mathbf{y}_{k'}, \nu') \left(1 \wedge \frac{\pi(k')}{\pi(k)}\right)$$

$$+ \delta_{(k, \mathbf{x}_k, -\nu)}(k', \mathbf{y}_{k'}, \nu') \left( 1 - 1 \wedge \frac{\pi(k + \nu)}{\pi(k)} \right).$$

Use  $\{(K, \mathbf{X}_K, \nu)_{\text{ideal}}(m) : m \in \mathbb{N}\}$  to denote the corresponding Markov chain. The transitions in the ideal case are therefore such that with probability  $1 \wedge \pi(k + \nu)/\pi(k)$  there is a move to model  $k + \nu$  with parameters  $\mathbf{y}_{k+\nu} \sim \pi(\cdot \mid k + \nu)$  (and the direction  $\nu$  is conserved). Otherwise, the model and parameters stay the same (and the direction  $\nu$  is reversed). The two distinctive elements of the ideal sampler are the form of the acceptance probability and distribution of the proposal  $\mathbf{y}_{k+\nu}$ . Intuitively, if [Algorithm 2](#) proposes parameters with a distribution close to  $\pi(\cdot \mid k + \nu)$  and accept them with a probability close to  $1 \wedge \pi(k + \nu)/\pi(k)$  (in the limit), the weak convergence should happen as the transition probabilities share the same behaviour. This is essentially what Theorem 1 in [Karr \(1975\)](#) indicates: if [Algorithm 2](#) and its ideal version are initialised in the same way (i.e.  $(K, \mathbf{X}_K, \nu)_T(0)$  and  $(K, \mathbf{X}_K, \nu)_{\text{ideal}}(0)$  follow the same distribution), and  $P_T \rightarrow P_{\text{ideal}}$  in some sense as  $T \rightarrow \infty$ , then  $\{(K, \mathbf{X}_K, \nu)_T(m) : m \in \mathbb{N}\}$  converges weakly towards  $\{(K, \mathbf{X}_K, \nu)_{\text{ideal}}(m) : m \in \mathbb{N}\}$ , denoted by  $\{(K, \mathbf{X}_K, \nu)_T(m) : m \in \mathbb{N}\} \Rightarrow \{(K, \mathbf{X}_K, \nu)_{\text{ideal}}(m) : m \in \mathbb{N}\}$ , as  $T \rightarrow \infty$ .

We already know that the acceptance probabilities are the same in the limit for both samplers as it is mentioned in [Karagiannis and Andrieu \(2013\)](#) that  $r_{\text{NRJ2}}((k, \mathbf{x}_k), (k + \nu, \mathbf{y}_{k+\nu}^{(T-1)}))$  is a consistent estimator of  $\pi(k + \nu)/\pi(k)$  as  $T \rightarrow \infty$  under realistic assumptions. For our result, we more precisely consider the following assumption.

**Assumption 1.** *The random variable  $r_{\text{NRJ2}}((k, \mathbf{x}_k), (k + \nu, \mathbf{Y}_{k+\nu}^{(T-1)}))$  converges in distribution towards  $\pi(k + \nu)/\pi(k)$  as  $T \rightarrow \infty$ , for any given  $(k, \mathbf{x}_k, \nu)$ .*

The proposals for the parameters in [Algorithm 2](#)  $\mathbf{y}_{k+\nu}^{(T-1)}$  should in practice be distributed in the limit as  $\pi(\cdot \mid k + \nu)$ . Indeed, consider as in our practical example in [Section 5.2](#) and those in [Karagiannis and Andrieu \(2013\)](#) that  $K_{k \mapsto k+\nu}^{(t)}$  are  $\rho_{k \mapsto k+\nu}^{(t)}$ -reversible MH kernels in which the proposal distributions are the same for all  $t$ . This more precisely means that

$$\begin{aligned} K_{k \mapsto k+\nu}^{(t)}((\mathbf{y}_{k+\nu}^{(t-1)}, \mathbf{u}_{k+\nu \mapsto k}^{(t-1)}), (\mathbf{y}_{k+\nu}^{(t)}, \mathbf{u}_{k+\nu \mapsto k}^{(t)})) &:= q_{\text{NRJ2}}^{k, \nu}((\mathbf{y}_{k+\nu}^{(t-1)}, \mathbf{u}_{k+\nu \mapsto k}^{(t-1)}), (\mathbf{y}_{k+\nu}^{(t)}, \mathbf{u}_{k+\nu \mapsto k}^{(t)})) \\ &\times \left( 1 \wedge \frac{\rho_{k \mapsto k+\nu}^{(t)}(\mathbf{y}_{k+\nu}^{(t)}, \mathbf{u}_{k+\nu \mapsto k}^{(t)}) q_{\text{NRJ2}}^{k, \nu}((\mathbf{y}_{k+\nu}^{(t-1)}, \mathbf{u}_{k+\nu \mapsto k}^{(t-1)}), (\mathbf{y}_{k+\nu}^{(t)}, \mathbf{u}_{k+\nu \mapsto k}^{(t)}))}{\rho_{k \mapsto k+\nu}^{(t-1)}(\mathbf{y}_{k+\nu}^{(t-1)}, \mathbf{u}_{k+\nu \mapsto k}^{(t-1)}) q_{\text{NRJ2}}^{k, \nu}((\mathbf{y}_{k+\nu}^{(t)}, \mathbf{u}_{k+\nu \mapsto k}^{(t)}), (\mathbf{y}_{k+\nu}^{(t-1)}, \mathbf{u}_{k+\nu \mapsto k}^{(t-1)}))} \right) \end{aligned}$$

$$+ \delta_{(\mathbf{y}_{k+\nu}^{(t-1)}, \mathbf{u}_{k+\nu \mapsto k}^{(t-1)})}(\mathbf{y}_{k+\nu}^{(t)}, \mathbf{u}_{k+\nu \mapsto k}^{(t)}) \mathbb{P}_{q_{\text{NRJ2}}}(\text{rejection} \mid \mathbf{y}_{k+\nu}^{(t-1)}, \mathbf{u}_{k+\nu \mapsto k}^{(t-1)}),$$

where  $\mathbb{P}_{q_{\text{NRJ2}}}(\text{rejection} \mid \mathbf{y}_{k+\nu}^{(t-1)}, \mathbf{u}_{k+\nu \mapsto k}^{(t-1)})$  is the rejection probability starting from  $(\mathbf{y}_{k+\nu}^{(t-1)}, \mathbf{u}_{k+\nu \mapsto k}^{(t-1)})$  and using  $q_{\text{NRJ2}}^{k,\nu}$  as the proposal distribution (which is the same for all  $t$ ). If  $0 < t^* < T$  and  $T$  are such that  $t^*/T$  is close to 1, then  $\rho_{k \mapsto k+\nu}^{(t^*)}$  is essentially proportional to  $\pi(\cdot \mid k + \nu) \otimes q_{k+\nu \mapsto k}$  (see (6)), and this is true for all  $t \geq t^*$ . Therefore, the process associated to  $K_{k \mapsto k+\nu}^{(t)}$  with  $t \geq t^*$  is essentially a time-homogeneous Markov chain with  $\pi(\cdot \mid k + \nu) \otimes q_{k+\nu \mapsto k}$  as a stationary distribution. This is why if  $T$  is additionally such that  $T - t^*$  is large enough, then  $\mathbf{Y}_{k+\nu}^{(T-1)}$  is (approximately) distributed as  $\pi(\cdot \mid k + \nu)$ .

Consider  $\{(\mathbf{Y}_{k+\nu}, \mathbf{U}_{k+\nu \mapsto k})(m) : m \in \mathbb{N}\}$  to be the time-homogeneous  $\pi(\cdot \mid k + \nu) \otimes q_{k+\nu \mapsto k}$ -reversible Markov chain associated with the proposal distribution  $q_{\text{NRJ2}}^{k,\nu}$  (thus generated by a regular MH algorithm with the same proposal distribution  $q_{\text{NRJ2}}^{k,\nu}$  for all iteration  $m$  with a stationary distribution that is fixed and set to be  $\pi(\cdot \mid k + \nu) \otimes q_{k+\nu \mapsto k}$ ). The design of  $q_{\text{NRJ2}}^{k,\nu}$  has an impact on how large the distance between  $T$  and  $t^*$  need to be to have  $\mathbf{Y}_{k+\nu}^{(T-1)}$  approximately distributed as  $\pi(\cdot \mid k + \nu)$ . In our weak convergence result, we assume to simplify that it is such that the associated Markov chain is uniformly ergodic. It is highlighted in the proof what modifications and which additional technical conditions are required if geometric ergodicity is instead assumed.

**Assumption 2.** *For all  $k$  and  $\nu$ , the time-homogeneous  $\pi(\cdot \mid k + \nu) \otimes q_{k+\nu \mapsto k}$ -reversible Markov chain associated with the proposal distribution  $q_{\text{NRJ2}}^{k,\nu}$ ,  $\{(\mathbf{Y}_{k+\nu}, \mathbf{U}_{k+\nu \mapsto k})(m) : m \in \mathbb{N}\}$ , is uniformly ergodic.*

Finally, we assume regularity conditions on the PDF  $q_{\text{NRJ2}}^{k,\nu}$ .

**Assumption 3.** *For all  $k$  and  $\nu$ ,  $q_{\text{NRJ2}}^{k,\nu}((\mathbf{y}_{k+\nu}^{(t-1)}, \mathbf{u}_{k+\nu \mapsto k}^{(t-1)}), (\mathbf{y}_{k+\nu}^{(t)}, \mathbf{u}_{k+\nu \mapsto k}^{(t)}))$  and*

$$\frac{q_{\text{NRJ2}}^{k,\nu}((\mathbf{y}_{k+\nu}^{(t)}, \mathbf{u}_{k+\nu \mapsto k}^{(t)}), (\mathbf{y}_{k+\nu}^{(t-1)}, \mathbf{u}_{k+\nu \mapsto k}^{(t-1)}))}{q_{\text{NRJ2}}^{k,\nu}((\mathbf{y}_{k+\nu}^{(t-1)}, \mathbf{u}_{k+\nu \mapsto k}^{(t-1)}), (\mathbf{y}_{k+\nu}^{(t)}, \mathbf{u}_{k+\nu \mapsto k}^{(t)}))} \quad (10)$$

*are bounded above by a positive constant that depends only on  $k$  and  $\nu$ .*

Note that (10) is equal to 1 if  $q_{\text{NRJ2}}^{k,\nu}$  is symmetric. We are now ready to present the weak convergence result.

**Theorem 1** (Weak convergence of [Algorithm 2](#)). *Under Assumptions 1 to 3 and assuming that  $(K, \mathbf{X}_K, \nu)_T(0) \sim \pi \otimes \mathcal{U}\{-1, 1\}$  and  $(K, \mathbf{X}_K, \nu)_{ideal}(0) \sim \pi \otimes \mathcal{U}\{-1, 1\}$ , we have*

$$\{(K, \mathbf{X}_K, \nu)_T(m) : m \in \mathbb{N}\} \implies \{(K, \mathbf{X}_K, \nu)_{ideal}(m) : m \in \mathbb{N}\} \quad \text{as } T \longrightarrow \infty.$$

## 4 About NRJ performance

We provide in [Section 4.1](#) a theoretical result showing that the proposed NRJ samplers yield ergodic averages of lower asymptotic variance than the corresponding RJ samplers proposing uniformly at random either model  $k - 1$  or model  $k + 1$  if a model switch is attempted. Empirically, the level of improvement of NRJ over RJ increases as the samplers better approximate the ideal ones. We show in [Section 4.2](#) that the level of improvement also depends on the shape of the target. We finish in [Section 4.3](#) with a discussion about scaling limit results formalising a sharp divide in the exploration behaviour of NRJ versus RJ.

### 4.1 Asymptotic variance of ergodic averages

A corollary of Theorem 3.17 in [Andrieu and Livingstone \(2019\)](#) allows us to compare ergodic averages produced by  $P_{NRJ}$  with those produced by  $P_{RJ}^{unif}$ , where  $P_{NRJ}$  is the Markov kernel simulated by any NRJ (such as [Algorithm 1](#), [2](#) or [3](#)) and  $P_{RJ}^{unif}$  that simulated by its reversible counterpart with  $g(k, k + 1)/(1 - \tau) = g(k, k - 1)/(1 - \tau) = 1/2$  (representing conditional probabilities given that a model switch is proposed). This result establishes that the asymptotic variance of ergodic averages produced by  $P_{NRJ}$  is at most equal to that of ergodic averages produced by  $P_{RJ}^{unif}$  for bounded test functions. In particular, the estimates of posterior model probabilities have a lower asymptotic variance under  $P_{NRJ}$  than under  $P_{RJ}^{unif}$ .

**Corollary 1.** *For any real-valued bounded function  $f$  of  $(k, \mathbf{x}_k)$  considered without loss of generality to have zero mean under the target,*

$$var_\lambda(f, P_{NRJ}) \leq var_\lambda(f, P_{RJ}^{unif}),$$

where  $\text{var}_\lambda(f, P) := \mathbb{E}[\{f(K(0), \mathbf{X}_K(0))\}^2] + 2 \sum_{m>0} \lambda^m \mathbb{E}[f(K(0), \mathbf{X}_K(0))f(K(m), \mathbf{X}_K(m))]$  of  $\{(K(m), \mathbf{X}_K(m)) : m \in \mathbb{N}\}$  being a Markov chain of transition kernel  $P$  at equilibrium and  $\lambda \in [0, 1)$ . If  $P_{\text{NRJ}}$  is uniformly ergodic,  $\text{var}_\lambda(f, P_{\text{NRJ}})$  converges to the asymptotic variance  $\text{var}(f, P_{\text{NRJ}}) := \mathbb{E}[\{f(K(0), \mathbf{X}_K(0))\}^2] + 2 \sum_{m>0} \mathbb{E}[f(K(0), \mathbf{X}_K(0))f(K(m), \mathbf{X}_K(m))]$  as  $\lambda \rightarrow 1$ , and therefore,  $\text{var}(f, P_{\text{NRJ}}) \leq \text{var}(f, P_{\text{RJ}}^{\text{unif}})$  (the limit always exists for a reversible Markov chain).

We highlight in the proof of the corollary which additional technical condition is required for the limit to hold under geometric ergodicity.

## 4.2 Dependence on the shape of the target

We have shown that it is possible to construct samplers as close as we want to their ideal counterparts, at least in the weak convergence sense. We focus in the rest of the section on the marginal ideal behaviour of  $K$  associated with the ideal RJ and NRJ. We only consider iterations in which model switches are proposed to focus on this type of transitions. In particular we do not study the impact of the proportion of parameter updates  $\tau$ , but we discuss briefly how this parameter is selected in [Section 6.1](#).

In [Section 4.2.1](#), existing results describing the behaviours of ideal RJ and NRJ when the marginal distribution is uniform or log concave are presented. NRJ outperform RJ in the former case, but not necessarily in the latter if we consider using functions  $g$  incorporating information about this marginal distribution instead of the uniform as discussed earlier. To analyse this latter case further, we use a parameter  $\phi \geq 1$  to characterise log concave distributions in [Section 4.2.2](#), and present a family of “worst” (for NRJ) log concave distributions for which the larger is  $\phi$  the more concentrated is the PMF. NRJ outperform RJ when  $\phi$  is not too large and the target is a member of this family.

### 4.2.1 Existing results

Denote by  $\{K_{\text{ideal}}^{\text{RJ}, g}(m) : m \in \mathbb{N}\}$  and  $\{K_{\text{ideal}}^{\text{NRJ}}(m) : m \in \mathbb{N}\}$  the Markov chains produced by ideal RJ and NRJ, where we highlighted that the behaviour of RJ depends on  $g$ . We show here how this proposal distribution can impact performance.



We consider a scenario where  $\mathcal{K} := \{1, \dots, K_{\max}\}$ . When the target is uniform on this set, the process  $\{K_{\text{ideal}}^{\text{NRJ}}(m) : m \in \mathbb{N}\}$  evolves deterministically; all proposals are accepted and it thus goes from 1 to  $K_{\max}$  without stopping, and changes direction at  $K_{\max}$  to return to 1. It is thus periodic and the distribution of  $K_{\text{ideal}}^{\text{NRJ}}(m)$  does not converge towards the target as  $m \rightarrow \infty$ . This is however not an issue when approximating expectations with respect to the target. A randomised version of  $\{K_{\text{ideal}}^{\text{NRJ}}(m) : m \in \mathbb{N}\}$  exists, see [Diaconis et al. \(2000\)](#)<sup>2</sup>. These authors prove that their process also explores the space in  $\mathcal{O}(K_{\max})$  steps while it takes  $\mathcal{O}(K_{\max}^2)$  for  $\{K_{\text{ideal}}^{\text{RJ}, g^*}(m) : m \in \mathbb{N}\}$ ,  $g^*$  being the optimal proposal distribution. The usual symmetric distribution  $g^*(k, k+1) = g^*(k, k-1) = 1/2$  is the optimal (conditional) proposal distribution (given that a model switch is proposed) in this case among all symmetric stochastic tridiagonal matrices ([Boyd et al., 2006](#)); i.e. when one restricts oneself to proposals of the form  $k \mapsto k' \in \{k-1, k+1\}$ . This thus establishes the superiority of NRJ over RJ in this case among samplers with proposals of the form  $k \mapsto k' \in \{k-1, k+1\}$ .

Superiority for uniform targets is an interesting theoretical result, but this is not a scenario of interest in Bayesian model selection. We believe that it is more likely that the posterior distributions reflect a balance between too simple models (that are more stable but do not capture well the dynamics in the data) and too complex models (that overfit and have less generalisation power), in the spirit of Occam's razor. Unimodal distributions, which are such that  $\pi(1) \leq \dots \leq \pi(k^*) \geq \dots \geq \pi(K_{\max})$ , are in this sense more interesting to analyse. [Hildebrand \(2002\)](#) generalised the result of [Diaconis et al. \(2000\)](#) on the Markov chain similar to  $\{K_{\text{ideal}}^{\text{NRJ}}(m) : m \in \mathbb{N}\}$  to log concave distributions, defined as distributions such that  $\pi(k)/\pi(k-1) \geq \pi(k+1)/\pi(k)$  for all  $k \in \{2, \dots, K_{\max} - 1\}$ . Log concave distributions belong to the family of unimodal distributions. Indeed, if we consider for instance  $k > k^*$  (the mode), we observe that the ratios  $\pi(k+1)/\pi(k)$  are smaller and smaller as we get further away from the mode.

An adaptation of the proof of [Hildebrand \(2002\)](#) allows us to prove that  $\mathcal{O}(K_{\max})$  steps are sufficient for  $\{K_{\text{ideal}}^{\text{NRJ}}(m) : m \in \mathbb{N}\}$  to traverse the state-space, if we assume that the

---

<sup>2</sup>The difference is that instead of systematically changing direction at 1 and  $K_{\max}$ , the sampler changes direction probabilistically after on average  $K_{\max}$  steps, making it aperiodic.

distribution is log concave, but not uniform. For  $\{K_{\text{ideal}}^{\text{RJ},g}(m) : m \in \mathbb{N}\}$ , no such results are available. To establish the superiority of NRJ when the target is not too concentrated, we need to identify the optimal proposal distribution  $g^*$  for RJ and to prove that the number of required steps is larger. We take here a step in this direction.

We choose the competitor to NRJ to be the RJ with the distribution  $g^*$  given by

$$g^*(k, k') \propto \sqrt{\pi(k')/\pi(k)} \quad \text{for } k' \in \{k-1, k+1\}. \quad (11)$$

This choice finds its justification in [Zanella \(2020\)](#), in which it is shown that a class of what the author calls *informed* distributions with  $g^*$  as a special case are optimal within reversible samplers in some situations. In fact, it is possible to numerically show that the optimal distribution in terms of speed of convergence among distributions  $g(k, \cdot)$  defined on  $\{k-1, k+1\}$  is very close to  $g^*$  with a negligible speed difference when the log concave distribution belongs to the family defined in the next section. Note that the optimal proposal distribution  $g^*(k, k+1) = g^*(k, k-1) = 1/2$  is retrieved when the target is uniform.

For any log concave target,  $g^*$  is such that

$$\frac{1}{\pi(k)/\pi(k+1) + 1} \leq g^*(k, k+1) \leq \frac{1}{\pi(k-1)/\pi(k) + 1}.$$

The acceptance probabilities are controlled in the same way by ratios of posterior model probabilities. As long as the target is not too concentrated, meaning ratios not too far from 1,  $\{K_{\text{ideal}}^{\text{RJ},g^*}(m) : m \in \mathbb{N}\}$  thus still has a diffusive behaviour that makes it traverse the state-space in of order of  $K_{\text{max}}^2$  steps. However, for concentrated targets,  $g^*(k, k+1)$  gets close to 1 when the chain is at the left of the mode as  $\pi(k)/\pi(k+1)$  and  $\pi(k-1)/\pi(k)$  are close to 0. The stochastic process thus moves persistently towards the mode and wanders around it afterwards.

**Remark 1.** [Diaconis et al. \(2000\)](#), and afterwards [Hildebrand \(2004\)](#), also studied V-shaped distributions, which are a class of multimodal distributions. They showed that under regularity conditions it takes on the order of  $K_{\text{max}}^2$  and  $K_{\text{max}}^2 \log K_{\text{max}}$  steps to converge towards the target for the non-reversible and reversible (with a uniform proposal distribution) samplers, respectively, suggesting that the non-reversible sampler makes the chain

move quicker from a mode to another than its reversible counterpart for some multimodal distributions.

#### 4.2.2 Log concave distributions: a worst case scenario

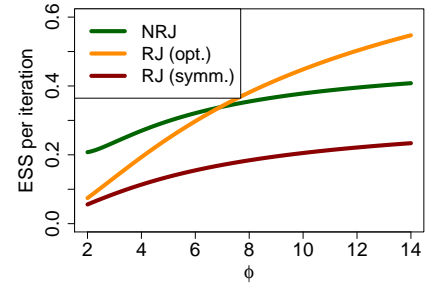
One way to characterise any log concave distribution is through the minimum of the ratios  $\pi(k-1)/\pi(k)$ , for  $k \leq k^*$ , and  $\pi(k+1)/\pi(k)$ , for  $k \geq k^*$ . Consider that this minimum is  $1/\phi$ . The distribution with a constant decreasing factor from the mode of  $1/\phi$  leads to RJ with  $g^*$  with the most significant advantage over RJ with a symmetric proposal. This is because the distribution is the most concentrated, which at the same time leaves not much room for persistent movement for NRJ.

We now introduce a class of distributions with this characteristic. This class is such that the mode  $k^*$  is at the middle of the domain:

$$\frac{\pi(k+1)}{\pi(k)} = \frac{1}{\phi} \quad \text{for } k \geq k^*, \quad \text{and} \quad \frac{\pi(k-1)}{\pi(k)} = \frac{1}{\phi} \quad \text{for } k \leq k^*, \quad \text{with } \phi > 1. \quad (12)$$

Numerically, if we set the target to be the distribution in (12), we observe in Figure 3

that NRJ outperforms RJ in terms of ESS when the target is not too concentrated by a factor, for instance, up to 2.8 when  $K_{\max} := 11$ . The concentration threshold is in this case around  $\phi = 7$ ; beyond this the target is too concentrated and RJ with  $g^*$  slowly starts to perform better.



Beyond this threshold, RJ with  $g^*$  is more efficient because there are basically 3 possible values for  $K$ : the mode  $k^*$ ,  $k^* + 1$  and  $k^* - 1$ . Indeed, when  $\phi$  is exactly 7, the total mass outside of these values is  $2 \sum_{k=k^*+2}^{K_{\max}} \pi(k^*)/\phi^{k-k^*} \approx 3.57\%$ . This is essentially true for any value of  $K_{\max}$  as this percentage is equal to the limiting value (to two decimal places) as  $K_{\max} \rightarrow \infty$ . When there are 3 possible values, starting from the mode  $k^*$ , both NRJ and RJ with  $g^*$  go either to the right or the left with equal probability (on average for NRJ given that  $\nu \sim \mathcal{U}\{-1, 1\}$ ). Let us say that they go to  $k^* + 1$ . The difference is that NRJ tries to go to  $k^* + 2$  (because of the direction), which is likely to be rejected, and therefore, it stays for one iteration at  $k^* + 1$ ; RJ directly goes back to  $k^*$  given that

$g^*(k^* + 1, k^*) = \phi/(\phi + 1)$  is close to 1. RJ thus seems to have an advantage in terms of required number of steps to traverse the state-space.

To summarise, NRJ is expected to perform better than RJ with  $g^*$  for any log concave distribution such that the minimum of the ratios  $\pi(k - 1)/\pi(k)$  and  $\pi(k + 1)/\pi(k)$  is larger than  $1/\phi^*$ , where  $\phi^* \approx 7$ . We noticed that  $g^*$  uses information about the target which is obviously not available prior pilot runs. Given that NRJ always outperform RJ with a symmetric proposal ([Corollary 1](#)), we thus recommend as practical guidelines to start by using NRJ, and if after pilot runs the target appears strongly concentrated, then it may be beneficial to switch to RJ with  $g^*$ . In the multiple change-point example in [Section 5.2](#), the target is for instance not too concentrated and RJ with  $g^*$  performs similarly to RJ with the symmetric proposal.

### 4.3 Scaling limits of model indicator process

Another way to evaluate the performance of algorithms is through the identification and analysis of scaling limits of their associated stochastic processes as the dimension  $d$  of the state-space goes to infinity. [Roberts et al. \(1997\)](#) and [Roberts and Rosenthal \(1998\)](#) applied this strategy to optimally tune the random walk Metropolis (RWM) and Metropolis-adjusted Langevin algorithm (MALA), but their analyses can also be used to establish that MALA is more efficient than RWM. This follows from the fact that to obtain non-trivial continuous limiting stochastic processes we need to speed up time by factors  $d$  and  $d^{1/3}$  for RWM and MALA, respectively. We explore such scaling limits for the processes  $\{K_{\text{ideal}}^{\text{RJ},g}(m) : m \in \mathbb{N}\}$  and  $\{K_{\text{ideal}}^{\text{NRJ}}(m) : m \in \mathbb{N}\}$ .

In our framework, we have no guarantee that the model indicator variable will converge towards a continuous random variable as  $K_{\text{max}}$  increases. In the supplementary material, we present strong and technical assumptions on  $\pi(k)$  under which results analogous to those of [Syed et al. \(2019\)](#) are obtained: the reversible process suitably rescaled converges to a diffusion while the non-reversible version converges to a piecewise-deterministic Markov process (see Theorems 2 and 3 in Section 2 of the supplementary material). The required time rescalings lead to conclusions consistent with the results presented in the previous

section showing that  $\mathcal{O}(K_{\max}^2)$  and  $\mathcal{O}(K_{\max})$  steps are required to explore the state-space for  $\{K_{\text{ideal}}^{\text{RJ},g}(m) : m \in \mathbb{N}\}$  and  $\{K_{\text{ideal}}^{\text{NRJ}}(m) : m \in \mathbb{N}\}$ , respectively.

## 5 Numerical experiments

Recall that in the usual non-ideal situation, the acceptance ratio in  $\alpha_{\text{NRJ}}$  can be viewed as the ideal ratio  $\pi(k')/\pi(k)$  corrupted by some multiplicative noise; see (5). In practice, the noise fluctuates around 1. In [Section 5.1](#), we show how the difference in performance between NRJ and RJ varies when the noise amplitude changes (in a sense made precise in that section), or in other words as we move away or towards ideal NRJ and RJ. The methods presented in [Section 3](#) are then applied to illustrate how their beneficial effect translates in practice for different noise behaviours. We also show how performances vary when the total number of models increases on a simple target distribution for which we can precisely control the noise behaviour and number of models. In [Section 5.2](#), we evaluate the performance of NRJ and RJ in a real multiple change-point problem.

### 5.1 Simulation study

Let the target distribution be

$$\pi(k, \mathbf{x}_k) = p_{\phi, K_{\max}}(k) \prod_{i=1}^k \varphi(x_{i,k}),$$

where  $p_{\phi, K_{\max}}$  is the PMF defined in [Section 4.2.2](#) in (12),  $\varphi$  is the density of a standard normal and  $\mathbf{x}_k := (x_{1,k}, \dots, x_{k,k})$ . When switching from model  $k$  to model  $k+1$  in this case, one parameter needs to be added. It is not necessary to move the parameters that were in model  $k$  given that they have the same distributions as the first  $k$  parameters of model  $k+1$ . In this context, it is straightforward to specify the functions  $T_{k \rightarrow k+1}$  that are required for the implementation of RJ and NRJ: they are such that the proposals for the parameters of model  $k+1$  are  $\mathbf{y}_{k+1} = (\mathbf{x}_k, u_{k \rightarrow k+1})$ . This also defines the functions  $T_{k+1 \rightarrow k}$  for the (deterministic) reverse moves. Note that  $\mathbf{u}_{k+1 \rightarrow k} = \emptyset$  for all  $k$ .

We have  $\pi(k+1)/\pi(k) = p_{\phi, K_{\max}}(k+1)/p_{\phi, K_{\max}}(k)$ , and the noise term  $\varepsilon$  is given by

$$\frac{\pi(\mathbf{y}_{k+1} \mid k+1) q_{k+1 \rightarrow k}(\mathbf{u}_{k+1 \rightarrow k})}{\pi(\mathbf{x}_k \mid k) q_{k \rightarrow k+1}(\mathbf{u}_{k \rightarrow k+1}) |J_{T_{k \rightarrow k+1}}(\mathbf{x}_k, \mathbf{u}_{k \rightarrow k+1})|^{-1}} = \frac{\varphi(u_{k \rightarrow k+1})}{q_{k \rightarrow k+1}(u_{k \rightarrow k+1})}. \quad (13)$$

We can therefore precisely control the noise behaviour by setting  $q_{k \rightarrow k+1} = \mathcal{N}(0, \sigma^2)$ , where  $\sigma > 0$  is the varying parameter. Indeed, in this case

$$\frac{\varphi(u_{k \rightarrow k+1})}{q_{k \rightarrow k+1}(u_{k \rightarrow k+1})} = \sigma \exp \left[ -\frac{u_{k \rightarrow k+1}^2}{2} \left( 1 - \frac{1}{\sigma^2} \right) \right],$$

which behaviour varies with  $\sigma$  given that  $u_{k \rightarrow k+1} \sim \mathcal{N}(0, \sigma^2)$ . This is also true for the reverse move. A small  $\sigma$  represents a proposal distribution that is more concentrated around the mode than the target, whereas it is less concentrated when  $\sigma$  is large.

For implementing [Algorithm 2](#) or the corresponding RJ, we only need to create paths for the proposals  $u_{k \rightarrow k+1}$  used to switch from model  $k$  to model  $k+1$ . This is realised by looking at the noise term (13), and also by remembering that it is not necessary to move the parameters that were in model  $k$ . We thus essentially create a bridge between model  $k$  to model  $k+1$  made of weighted geometric averages of  $f$  and  $q_{k \rightarrow k+1}$ . The annealing intermediate distributions indeed have intuitive forms:

$$\rho_{k \rightarrow k+1}^{(t)}(u_{k \rightarrow k+1}^{(t)}) \propto \exp \left( -\frac{(u_{k \rightarrow k+1}^{(t)})^2}{2} [(1 - \gamma_t)\sigma^{-2} + \gamma_t] \right),$$

where  $\gamma_t := t/T$ . Therefore, to go from model  $k$  to model  $k+1$ , we target normal distributions with mean 0 and variances  $[(1 - t/T)\sigma^{-2} + t/T]^{-1}$ ; we thus start with variances close to  $\sigma^2$  (corresponding to the initial proposal distribution) to finish with variances close to 1 (the target distribution). For the reverse move, we do the opposite. As we can sample from the distributions  $\rho_{k \rightarrow k+1}^{(t)}$ , we use them as transitions kernels:  $K_{k \rightarrow k+1}^{(t)}(u_{k \rightarrow k+1}^{(t)}, \cdot) := \rho_{k \rightarrow k+1}^{(t)}$  (which satisfy the symmetry (8) and reversibility (9) conditions).

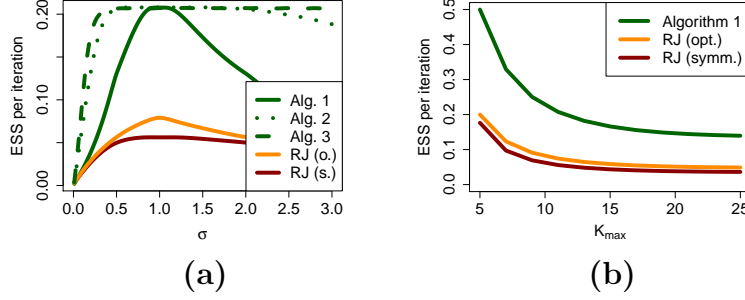
The results are presented in [Figure 4](#). They are based on 1,000 runs of 100,000 iterations for each value of  $\sigma$  and  $K_{\max}$ ; recall that the impact of varying  $\phi$  was discussed in [Section 4](#). As expected, the further  $\sigma$  is from 1 (the latter corresponding to ideal samplers), the lower is the ESS. We notice that the impact is almost symmetric in  $\sigma$  if we consider distances from the distribution  $\mathcal{N}(0, 1)$  (e.g. the normal with  $\sigma = 1/2$  is two times more concentrated, whereas the normal with  $\sigma = 2$  is two times less concentrated; they both are at a distance

of two). We also notice that in extreme cases, for instance when  $\sigma$  is close to 0, NRJ and RJ have similar performances. This is explained by the fact that the direction-assisted scheme characterising NRJ does not help any more; almost all moves are rejected, which implies that direction changes very often. This leads to the same diffusive behaviour as RJ. Applying Algorithms 2 and 3 improve performances. It is possible to obtain essentially flat lines around the maximum value of 0.21 ESS per iteration by increasing  $T$  and  $N$ , leading to samplers that are at least 2.5 times more efficient than RJ for any value of  $\sigma$ . Note that we do not show the results for the RJ corresponding to Algorithms 2 and 3 as it does not add information to Figure 4 (a) given that the lines would be on top of each other.

The ESS also decreases as the total number of models  $K_{\max}$  increases (see Figure 4 (b)). This is expected as the difference between  $k$  and  $k'$  (representing the current model and the next one to explore) is constant, equal to 1. The exploration abilities of the stochastic processes thus diminish as a smaller fraction of the state-space is traversed at each iteration. In theory, a way to compensate is to generate a random variable at each iteration that dictates the difference between  $k$  and  $k'$ , allowing larger jumps. However, as mentioned in Section 2.1, proposal distributions for transitions to models at a distance of more than 1 are often very difficult to design. Note that the total probability mass of the 15 most likely models is essentially 1 when  $\phi := 2$  (and  $K_{\max} \geq 15$ ), which explains why the ESS becomes constant beyond this value. Note also that we do not show the results for Algorithms 2 and 3 as  $\sigma := 1$ , meaning that ideal samplers are applied.

## 5.2 Performance evaluation in multiple change-point problems

In this section, we evaluate the performance of RJ and NRJ algorithms when applied to sample from the posterior of the model in Green (1995) for multiple change-point analysis, based on the coal mining disaster data set detailed in Raftery and Akman (1986). The  $n$  data points  $\mathbf{t} := (t_1, \dots, t_n)$  represent times of occurrence of disasters. It is assumed that they arose from a non-homogeneous Poisson process that has an intensity given by a step function  $\lambda_k$  with  $k + 1$  steps, where  $k \in \mathcal{K} := \{0, \dots, K_{\max}\}$ ,  $K_{\max}$  being a known positive integer.



**Figure 4:** (a) ESS as a function of  $\sigma$  for NRJ (Algorithm 1, Algorithm 2 with  $T := 15$ , and Algorithm 3 with  $T := 15$  and  $N := 15$ ) and RJ (with optimal and symmetric  $g$ ), when  $\phi := 2$  and  $K_{\max} := 11$ ; (b) ESS as a function of  $K_{\max}$  for NRJ (Algorithm 1) and RJ (with optimal and symmetric  $g$ ), when  $\phi := 2$  and  $\sigma := 1$

We use the same prior distributions and the same proposals for the RJ and NRJ as Green (1995). For implementing Algorithm 3 and the corresponding RJ, we proceed as in Karagiannis and Andrieu (2013). All details are provided in Section 3 of the supplementary material.

The performance of the different algorithms are summarised in Table 1. The results for Algorithm 3 and the corresponding RJ are based on 1,000 runs with 100,000 iterations and burn-ins of 10,000. To reach the *same computational budget* as these samplers, Algorithm 1 and its reversible counterpart are run with an increased number of iterations. The performance of ideal samplers with the same run length as Algorithm 3 is also presented in Table 1 to show the kind of performance that can be achieved. To measure performance, we display ESS per iteration. We also use the relative difference in total variation (TV) with the ideal NRJ:  $(\text{TV}(P) - \text{TV}(P_{\text{ideal}}^{\text{NRJ}})) / \text{TV}(P_{\text{ideal}}^{\text{NRJ}})$ , where  $\text{TV}(P)$  is the TV between the model distribution estimated using the Markov kernel  $P$  and the posterior model probabilities.<sup>3</sup>

We observe for this multiple change-point example that NRJ samplers always perform better the corresponding RJ samplers at no additional computational cost both in terms of relative TV error and ESS per iteration. Additionally, the displayed relative difference

<sup>3</sup>We used accurate approximations to the posterior model probabilities. We verified that the TV goes to 0 for all algorithms as the number of iterations increases.



in TV is obtained by running the vanilla samplers for the same amount of compute time as [Algorithm 3](#) and its corresponding RJ. It is thus clear that the vanilla samplers provide estimates of the marginal posterior model probabilities which are inaccurate. This is consistent with previous experimental results in [Karagiannis and Andrieu \(2013\)](#). To summarise, this example illustrates that, *at fixed computational complexity*, [Algorithm 3](#) is an algorithm that can outperform vanilla samplers and the RJ schemes proposed in [Karagiannis and Andrieu \(2013\)](#) and [Andrieu et al. \(2018\)](#).

Algorithms	Rel. diff. in TV	ESS per it.
Ideal NRJ	—	0.35
Ideal RJ	0.94	0.09
<a href="#">Algorithm 3</a>	0.94	0.15
Corresponding RJ	1.50	0.07
Vanilla NRJ	15.76	0.02
Vanilla RJ	16.66	0.01

**Table 1:** Performance of vanilla samplers (i.e. [Algorithm 1](#) and the corresponding RJ), [Algorithm 3](#) with  $T = 100$  and  $N = 10$  and the corresponding RJ, and ideal samplers

## 6 Discussion

In this paper, we have introduced non-reversible trans-dimensional samplers that can be applied to Bayesian nested model selection. They are derived from RJ algorithms by making simple modifications which require no additional computational cost during implementation; the model indicator process now follows a direction  $\nu$  which is conserved as long as the model switches are accepted, but reversed at the next rejection. Empirically, these samplers outperform their reversible counterparts when the marginal posterior distribution of  $K$  is not too concentrated. We now discuss some implementation aspects that have not been addressed in previous sections and possible directions for future research.

## 6.1 Other implementation aspects

Several functions need to be specified for implementing trans-dimensional samplers:  $g$  (which corresponds to the specification of  $\tau$  for NRJ),  $q_{k \mapsto k'}$  and  $T_{k \mapsto k'}$ . Significant amount of work has been carried out to address the specification of the last two when no prior information about the problems can be exploited (contrary to the examples in [Section 5](#)) or a more automatic perspective is adopted (see, e.g., [Green \(2003\)](#) and [Brooks et al. \(2003\)](#)). The approaches of these authors are arguably the most popular. They are directly applicable in the NRJ framework. We believe a particularly good way to proceed is to design the functions  $q_{k \mapsto k'}$  and  $T_{k \mapsto k'}$  according to the approach of [Green \(2003\)](#) to afterwards use them in [Algorithm 3](#) to benefit from the strategies of [Karagiannis and Andrieu \(2013\)](#) and [Andrieu et al. \(2018\)](#) that aim to ensure good mixing properties.

Little attention has been devoted to the impact of the specification of  $\tau$ . The choice and tuning of this parameter representing the proportion of parameter updates during an algorithm run is a non-trivial problem common to all trans-dimensional samplers whose solution depends on their ability at sampling both the parameters and model indicator. [Gagnon et al. \(2019\)](#) essentially devoted a whole paper on its impact on a specific reversible jump sampler's outputs. For a fixed computational budget, a value closer to 1 leads to more accurate parameter estimation (of the visited models), while a value closer to 0 yields better posterior model probability approximations. Studying more precisely the quantitative impact of  $\tau$  on the performance of the samplers is beyond the scope of this paper. However, from a qualitative point of view,  $\tau$  has no impact on the order between the asymptotic variances of NRJ and RJ; i.e. [Corollary 1](#) holds whatever being  $\tau$ .

[Gagnon et al. \(2019\)](#) prove weak convergence results for RJ under strong assumptions and identify ranges of values for which a suitable balance between a lot of model switches (but few parameter updates) and a lot of parameter updates (but few model switches) is reached. Values around 0.4 are suitable in the situation where  $q_{k \mapsto k'}$  and  $T_{k \mapsto k'}$  are well designed; otherwise, smaller values should be used. In scenarios in which NRJ is better at sampling  $K$  than RJ, it is expected that larger values for  $\tau$  than in RJ would be suitable. Further investigations are however required.

## 6.2 Possible directions for future research

We identified in [Section 4](#) a specific ideal RJ (associated with  $g^*$ ) as the main competitor to NRJ within all ideal RJ algorithms when the marginal posterior distribution of  $K$  belongs to a family of unimodal PMF and samplers are restricted to model switching proposals of the form  $k \mapsto k' \in \{k - 1, k + 1\}$ . We next provided arguments explaining why ideal NRJ outperform this ideal RJ when the target is not too concentrated and numerically showed the range of concentration parameters  $\phi$  in the PMF [\(12\)](#) for which this is the case. It would be interesting to conduct an exhaustive theoretical analysis to expand the scope of the conclusions and make more precise the expected gain.

It would also be interesting to develop NRJ that can be applied to non-nested models. However, developing efficient non-reversible samplers in such scenarios is much more difficult because, contrary to the nested case, there is no natural order among the models.

## Acknowledgements

The authors thank three anonymous referees for helpful suggestions that led to an improved paper. Philippe Gagnon acknowledges support from FRQNT (Le Fonds de recherche du Québec - Nature et technologies). Arnaud Doucet was partially supported by the U.S. Army Research Laboratory and the U. S. Army Research Office, and by the U.K. Ministry of Defence (MoD) and the U.K. Engineering and Physical Research Council (EPSRC) under grant number EP/R013616/1. He is also supported by the EPSRC grants EP/R018561/1 and EP/R034710/1.

## References

- Andrieu, C., A. Doucet, S. Yildirim, and N. Chopin (2018). On the utility of Metropolis–Hastings with asymmetric acceptance ratio. *arXiv:1803.09527*.
- Andrieu, C. and S. Livingstone (2019). Peskun-Tierney ordering for Markov chain and process Monte Carlo: beyond the reversible scenario. *arXiv:1906.06197*.

- Boyd, S., P. Diaconis, J. Sun, and L. Xiao (2006). Fastest mixing Markov chain on a path. *The American Mathematical Monthly* 113(1), 70–74.
- Brooks, S. P., P. Giudici, and G. O. Roberts (2003). Efficient construction of reversible jump Markov chain Monte Carlo proposal distributions. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* 65(1), 3–39.
- Chen, F., L. Lovász, and I. Pak (1999). Lifting Markov chains to speed up mixing. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pp. 275–281.
- Diaconis, P., S. Holmes, and R. M. Neal (2000). Analysis of a nonreversible Markov chain sampler. *Ann. Appl. Probab.*, 726–752.
- Gagnon, P. (2019). A step further towards automatic and efficient reversible jump algorithms. arXiv:1911.02089.
- Gagnon, P., M. Bédard, and A. Desgagné (2019). Weak convergence and optimal tuning of the reversible jump algorithm. *Math. Comput. Simulation* 161, 32–51.
- Gagnon, P., M. Bédard, and A. Desgagné (2020). An automatic robust Bayesian approach to principal component regression. *Journal of Applied Statistics*, 1–21.
- Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika* 82(4), 711–732.
- Green, P. J. (2003). Trans-dimensional Markov chain Monte Carlo. In *Highly structured stochastic systems*, pp. 179–196. OXFORD UNIV PRESS.
- Hildebrand, M. (2002). Analysis of the Diaconis-Holmes-Neal Markov chain sampler for log concave probabilities. <https://www.albany.edu/~martinhi/dvifiles/dhnlc4.dvi>.
- Hildebrand, M. (2004). Rates of convergence of the Diaconis-Holmes-Neal Markov chain sampler with a V-shaped stationary probability. *Markov. Process. Relat. Fields* 10, 687–704.

- Karagiannis, G. and C. Andrieu (2013). Annealed importance sampling reversible jump MCMC algorithms. *J. Comp. Graph. Stat.* 22(3), 623–648.
- Karr, A. F. (1975). Weak convergence of a sequence of Markov chains. *Z. Wahrsch. Verw. Gebiete* 33(1), 41–48.
- Neal, R. M. (2001). Annealed importance sampling. *Stat. Comput.* 11(2), 125–139.
- Neal, R. M. (2011). MCMC using Hamiltonian dynamics. In *Handbook of Markov Chain Monte Carlo*, pp. 113–160. CRC Press New York, NY.
- Raftery, A. E. and V. Akman (1986). Bayesian analysis of a Poisson process with a change-point. *Biometrika*, 85–89.
- Richardson, S. and P. J. Green (1997). On Bayesian analysis of mixtures with an unknown number of components. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* 59(4), 731–792.
- Roberts, G. O., A. Gelman, and W. R. Gilks (1997). Weak convergence and optimal scaling of random walk Metropolis algorithms. *Ann. Appl. Probab.* 7(1), 110–120.
- Roberts, G. O. and J. S. Rosenthal (1998). Optimal scaling of discrete approximations to Langevin diffusions. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* 60(1), 255–268.
- Sakai, Y. and K. Hukushima (2016). Irreversible simulated tempering. *J. Phys. Soc. Jpn.* 85(10), 104002.
- Syed, S., A. Bouchard-Côté, G. Deligiannidis, and A. Doucet (2019). Non-reversible parallel tempering: a scalable highly parallel MCMC scheme. *arXiv:1905.02939*.
- Vanetti, P., A. Bouchard-Côté, G. Deligiannidis, and A. Doucet (2017). Piecewise deterministic Markov chain Monte Carlo. *arXiv:1707.05296*.
- Vermaak, J., C. Andrieu, A. Doucet, and S. Godsill (2004). Reversible jump Markov chain Monte Carlo strategies for Bayesian model selection in autoregressive processes. *J. Time Series Anal.* 25(6), 785–809.

Vettori, S., R. Huser, J. Segers, and M. G. Genton (2019). Bayesian model averaging over tree-based dependence structures for multivariate extremes. *J. Comput. Graph. Statist.*, 1–17.

Zanella, G. (2020). Informed proposals for local MCMC in discrete spaces. *J. Amer. Statist. Assoc.* 115(530), 852–865.

## SUPPLEMENTARY MATERIAL

**Title:** Non-reversible jump algorithms for Bayesian nested model selection — Supplementary material (pdf).

**Section 1:** We present the proofs of [Proposition 1](#), [Theorem 1](#) and [Corollary 1](#).

**Section 2:** We present weak convergence results for the ideal samplers as the size of the state-space increases.

**Section 3:** The details about the multiple change-point example of [Section 5.2](#) are provided.