Non-reversible jump algorithms for Bayesian nested model selection

Philippe Gagnon and Arnaud Doucet

University of Oxford

January 8, 2020
Running example
1 Running example

2 Reversible jump samplers
   - Sampling context
   - Nested models
1. Running example

2. Reversible jump samplers
   - Sampling context
   - Nested models

3. Non-reversible jump samplers
   - Construction
   - Ideal NRJ
   - Towards ideal NRJ
   - Example (cont’d)
Running example

Reversible jump samplers
- Sampling context
- Nested models

Non-reversible jump samplers
- Construction
- Ideal NRJ
- Towards ideal NRJ
- Example (cont’d)
Example: multiple change-point problems

- Context: coal mining disasters.
Example: multiple change-point problems

- Context: coal mining disasters.
- Data: times of occurrence of disasters.
Example: multiple change-point problems

- Context: coal mining disasters.
- Data: times of occurrence of disasters.
- Model: non-homogeneous Poisson process.
Example: multiple change-point problems

- Context: coal mining disasters.
- Data: times of occurrence of disasters.
- Model: non-homogeneous Poisson process.
  - Assumption: the intensity is a step function.
Example: multiple change-point problems

- **Context:** coal mining *disasters*.
- **Data:** *times* of occurrence of disasters.
- **Model:** non-homogeneous *Poisson process*.
  - Assumption: the intensity is a *step function*.

### Context: coal mining disasters.

### Data: times of occurrence of disasters.

### Model: non-homogeneous Poisson process.

### Assumption: the intensity is a step function.
MH/fixed-dimensional MCMC sampler

Figure 2: Proposal $y$ with $\alpha(x, y) = 0.08$
1. Running example

2. Reversible jump samplers
   - Sampling context
   - Nested models

3. Non-reversible jump samplers
   - Construction
   - Ideal NRJ
   - Towards ideal NRJ
   - Example (cont’d)
Goal: sample from a target distribution $\pi(k, x_k)$,
Goal: sample from a target distribution $\pi(k, x_k)$,

$k \in \mathcal{K}$: model indicator,
Goal: sample from a target distribution $\pi(k, x_k)$,
- $k \in K$: model indicator,
- $x_k \in \mathbb{R}^{d_k}$: parameters of Model $k$, 
Goal: sample from a target distribution $\pi(k, x_k)$,
- $k \in \mathcal{K}$: model indicator,
- $x_k \in \mathbb{R}^{d_k}$: parameters of Model $k$,
- $d_k$: number of parameters in Model $k$. 
Reversible jump (RJ) algorithms

1. Generate $k' \sim g(k, \cdot)$.
2. Generate $u_k \mapsto k' \sim q_k \mapsto k'$.
3. Transform $(x_k, u_k \mapsto k')$: $T_k \mapsto k'(x_k, u_k \mapsto k') := (y_k', u_k' \mapsto k')$. If $u_a \sim U(0,1) \leq \alpha_{\text{RJ}}((k, x_k), (k', y_{k'})) := 1 \land \pi(k', y_{k'}) g(k', k) q_{k'} \mapsto k'(u_{k'} \mapsto k') | J_{T_k \mapsto k'}(x_k, u_k \mapsto k') | - 1$, set the next state of the chain to $(k', y_{k'})$. Otherwise, set it to $(k, x_k)$.
4. Go to Step 1.
Reversible jump (RJ) algorithms

1. Generate $k' \sim g(k, \cdot)$. 

Go to Step 1.
Reversible jump (RJ) algorithms

1. Generate $k' \sim g(k, \cdot)$.
2. Generate $u_{k \rightarrow k'} \sim q_{k \rightarrow k'}$.
Reversible jump (RJ) algorithms

1. Generate $k' \sim g(k, \cdot)$.
2. Generate $u_{k \mapsto k'} \sim q_{k \mapsto k'}$.
3. Transform $(x_k, u_{k \mapsto k'})$: $T_{k \mapsto k'}(x_k, u_{k \mapsto k'}) := (y_{k'}, u_{k' \mapsto k})$. If $u_a \sim \mathcal{U}(0, 1) \leq \alpha_{\text{RJ}}((k, x_k), (k', y_{k'}))$

$$:= 1 \wedge \frac{\pi(k', y_{k'}) g(k', k) q_{k' \mapsto k}(u_{k' \mapsto k})}{\pi(k, x_k) g(k, k') q_{k \mapsto k'}(u_{k \mapsto k'}) |J_{T_{k \mapsto k'}}(x_k, u_{k \mapsto k'})|^{-1}},$$

set the next state of the chain to $(k', y_{k'})$. Otherwise, set it to $(k, x_k)$.
4. Go to Step 1.
Example (cont’d)

Figure 3: Proposal \((k', y_{k'})\) with \(\alpha_{RJ}((k, x_k), (k', y_{k'})) = 1\)
Nested models

- We focus on nested models.
We focus on nested models.

$K$: ordinal discrete random variable reflecting the complexity of the models.
We focus on nested models.

$K$: ordinal discrete random variable reflecting the complexity of the models.

Examples:
We focus on nested models.

\( K \): ordinal discrete random variable reflecting the complexity of the models.

**Examples:**

1. number of change-points in multiple change-point problems,
Nested models

- We focus on nested models.
- $K$: ordinal discrete random variable reflecting the complexity of the models.
- **Examples:**
  1. number of change-points in multiple change-point problems,
  2. order of an autoregressive process,
Nested models

- We focus on nested models.
- \( K \): ordinal discrete random variable reflecting the complexity of the models.
- **Examples:**
  1. number of change-points in multiple change-point problems,
  2. order of an autoregressive process,
  3. number of components in mixture modelling.
RJ Algorithms (Cont’d)

Usually,

\[
g(k, k') = \begin{cases} 
\tau & \text{if } k' = k, \\
(1 - \tau)/2 & \text{if } k' = k - 1 \text{ or } k + 1.
\end{cases}
\]
Usually,

\[ g(k, k') = \begin{cases} 
\tau & \text{if } k' = k, \\
(1 - \tau)/2 & \text{if } k' = k - 1 \text{ or } k + 1.
\end{cases} \]

Remark: the model space \( \mathcal{K} \) is explored through a random walk.

\[ \text{ESS} = 0.04 \text{ per it.} \]
Construction of non-reversible jump (NRJ) algorithms

1. **Extend** the state space: add the set \{-1, +1\}.

<table>
<thead>
<tr>
<th>Running example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reversible jump samplers</td>
</tr>
<tr>
<td>Non-reversible jump samplers</td>
</tr>
<tr>
<td>Construction</td>
</tr>
<tr>
<td>Ideal NRJ</td>
</tr>
<tr>
<td>Towards ideal NRJ</td>
</tr>
<tr>
<td>Example (cont’d)</td>
</tr>
</tbody>
</table>
Construction of non-reversible jump (NRJ) algorithms

1. **Extend** the state space: add the set \{-1, +1\}.

   - Associate to it a **direction** variable \( \nu \sim \mathcal{U}\{-1, +1\} \) (new target = \( \pi \otimes \mathcal{U}\{-1, +1\} \)).
Construction of non-reversible jump (NRJ) algorithms

1. **Extend** the state space: add the set \([-1, +1]\).
   - Associate to it a **direction** variable \(\nu \sim U\{-1, +1\}\) (new target = \(\pi \otimes U\{-1, +1\}\)).

2. Set
   
   \[
   g(k, k') := \begin{cases} 
   \tau & \text{if } k' = k, \\
   1 - \tau & \text{if } k' = k + \nu.
   \end{cases}
   \]
Running example
Reversible jump samplers
Non-reversible jump samplers

Construction
Ideal NRJ
Towards ideal NRJ
Example (cont’d)

Construction of non-reversible jump (NRJ) algorithms

1. **Extend** the state space: add the set \{-1, +1\}.
   - Associate to it a **direction** variable \( \nu \sim \mathcal{U}\{-1, +1\} \) (new target = \( \pi \otimes \mathcal{U}\{-1, +1\} \)).

2. Set
   
   \[
   g(k, k') := \begin{cases} 
   \tau & \text{if } k' = k, \\
   1 - \tau & \text{if } k' = k + \nu.
   \end{cases}
   \]

3. **Define the acceptance probability** as:

   \[
   \alpha_{\text{NRJ}}((k, x_k), (k', y_{k'})) := 1 \wedge \frac{\pi(k', y_{k'}) q_{k' \rightarrow k}(u_{k' \rightarrow k})}{\pi(k, x_k) q_{k \rightarrow k'}(u_{k \rightarrow k'}) |J_{T_{k \rightarrow k'}}(x_k, u_{k \rightarrow k'})|^{-1}}.
   \]
Construction of non-reversible jump (NRJ) algorithms

1. **Extend** the state space: add the set \{-1, +1\}.
   - Associate to it a **direction** variable \(\nu \sim U\{-1, +1\}\) (new target = \(\pi \otimes U\{-1, +1\}\)).

2. **Set**
   \[
g(k, k') := \begin{cases} 
   \tau & \text{if } k' = k, \\
   1 - \tau & \text{if } k' = k + \nu.
   \end{cases}
   \]

3. **Define the acceptance probability** as:
   \[
   \alpha_{NRJ}((k, x_k), (k', y_{k'})) := 1 \wedge \frac{\pi(k', y_{k'}) q_{k'\rightarrow k}(u_{k'\rightarrow k})}{\pi(k, x_k) q_{k\rightarrow k'}(u_{k\rightarrow k'}) J_{T_{k\rightarrow k'}}(x_k, u_{k\rightarrow k'})^{-1}}.
   \]

4. **Accepted** proposals: set the next state to \((k', y_{k'}, \nu)\).
Construction of non-reversible jump (NRJ) algorithms

1. Extend the state space: add the set \{-1, +1\}.
   - Associate to it a direction variable \( \nu \sim \mathcal{U}\{-1, +1\}\) (new target = \( \pi \otimes \mathcal{U}\{-1, +1\}\)).

2. Set
   
   \[
   g(k, k') := \begin{cases} 
   \tau & \text{if } k' = k, \\
   1 - \tau & \text{if } k' = k + \nu. 
   \end{cases}
   \]

3. Define the acceptance probability as:
   
   \[
   \alpha_{NRJ}((k, x_k), (k', y_{k'})) := 1 \wedge \frac{\pi(k', y_{k'}) q_{k' \rightarrow k}(u_{k' \rightarrow k})}{\pi(k, x_k) q_{k \rightarrow k'}(u_{k \rightarrow k'}) J_{T_{k \rightarrow k'}}(x_k, u_{k \rightarrow k'})} |^{-1}. \]

4. Accepted proposals: set the next state to \((k', y_{k'}, \nu)\).
5. Rejected proposals: set the next state to \((k, x_k, -\nu)\).
NRJ VS RJ

RJ: ESS = 0.04 per it.

NRJ: ESS = 0.19 per it.

Figure 5: Trace plots for ideal RJ selecting uniformly at random which model to switch to and NRJ algorithms, and showing only the iterations in which model switches are proposed.
Ideal NRJ

- Ideal situation: $y_{k'} = u_{k \leftrightarrow k'} \sim q_{k \leftrightarrow k'} = \pi(\cdot | k')$, $\alpha_{NRJ}(\cdot)$.
Ideal NRJ

- **Ideal situation:** \( y_{k'} = u_{k \leftrightarrow k'} \sim q_{k \leftrightarrow k'} = \pi(\cdot | k') \),

- implying that

\[
\alpha_{\text{NRJ}}((k, x_k), (k', y_{k'})) = 1 \land \frac{\pi(k')}{\pi(k)}.
\]
Ideal situation: $y_{k'} = u_{k \rightarrow k'} \sim q_{k \rightarrow k'} = \pi(\cdot | k')$, implying that

$$\alpha_{NRJ}((k, x_k), (k', y_{k'})) = 1 \wedge \frac{\pi(k')}{\pi(k)}.$$

Typically:

$$\alpha_{NRJ}((k, x_k), (k', y_{k'})) = 1 \wedge \frac{\pi(k')}{\pi(k)} \varepsilon(x_k, y_{k'}).$$
Towards ideal NRJ

- Methods allow to get as close as we want to the ideal NRJ:
Towards ideal NRJ

- Methods allow to get as close as we want to the ideal NRJ:
  - [Karagiannis and Andrieu, 2013] and [Andrieu et al., 2018].
Towards ideal NRJ

- Methods allow to get as close as we want to the ideal NRJ:
  - [Karagiannis and Andrieu, 2013] and [Andrieu et al., 2018].

- How?
Towards ideal NRJ

- Methods allow to get as close as we want to the ideal NRJ:
  - [Karagiannis and Andrieu, 2013] and [Andrieu et al., 2018].

- How?
  - Sophisticated proposal schemes.
Towards ideal NRJ

Methods allow to get as close as we want to the ideal NRJ:

- [Karagiannis and Andrieu, 2013] and [Andrieu et al., 2018].

How?

- Sophisticated proposal schemes.
- By-product: $\varepsilon \longrightarrow 1$. 
Towards ideal NRJ

- Methods allow to get as close as we want to the ideal NRJ:
  - [Karagiannis and Andrieu, 2013] and [Andrieu et al., 2018].

- How?
  - Sophisticated proposal schemes.
  - By-product: $\varepsilon \to 1$.

- Fact: Markov chains simulated by NRJ converge weakly to those produced by ideal NRJ.
1. Running example

2. Reversible jump samplers
   - Sampling context
   - Nested models

3. Non-reversible jump samplers
   - Construction
   - Ideal NRJ
   - Towards ideal NRJ
   - Example (cont’d)
Example (cont’d)

(a) Vanilla samplers  (b) In between  (c) Ideal samplers

Figure 8: ESS for NRJ and RJ when the samplers are: (a) vanilla samplers, (b) in between samplers, (c) ideal samplers
Thank you for your attention.
Thank you

- Thank you for your attention.
- Questions?

On the utility of Metropolis–Hastings with asymmetric acceptance ratio.

*arXiv:1803.09527.*

Annealed importance sampling reversible jump MCMC algorithms.