Drzewa losowe i algorytmy Monte Carlo (Random trees and Monte Carlo algorithms)

Wojciech Niemiro[1]

Kolokwium Wydziałowe, MIMUW, październik 2019

[1] Mostly based on joint work with Tomasz Cąkała and Błażej Miasojedow (MIMUW).

IS, SMC, MCMC, PMCMC

Algorithms for sampling from (complicated) probability distributions:

Importance (Weighted) Sampling, IS

Markov Chain Monte Carlo, MCMC → **Particle MCMC**

Sequential Monte Carlo, SMC

Introduction: Simplified Example

$p(\cdot)$ – probability density on space $\mathcal{X}$. Let

$$\pi(x) = \frac{p(x)w(x)}{z}.$$

Introduction: Simplified Example

$p(\cdot)$ – probability density on space $\mathcal{X}$. Let

$$\pi(x) = \frac{p(x)w(x)}{z}.$$

Assume that:

- $p(\cdot)$ is easy to sample from.
- $w(\cdot) \geqslant 0$ is a weight function.
- $\pi(\cdot)$ is of interest (difficult *target* distribution).
- $z = \int p(x)w(x)\mathrm{d}x$ is a normalizing constant (intractable).

Introduction: Simplified Example

$p(\cdot)$ – probability density on space $\mathcal{X}$. Let

$$\pi(x) = \frac{p(x)w(x)}{z}.$$

Assume that:

- $p(\cdot)$ is easy to sample from.
- $w(\cdot) \geqslant 0$ is a weight function.
- $\pi(\cdot)$ is of interest (difficult *target* distribution).
- $z = \int p(x)w(x)\mathrm{d}x$ is a normalizing constant (intractable).

We can sample $X \sim p(\cdot)$ and compute $w(x)$ for a given $x \in \mathcal{X}$.

We want to sample from $\pi(\cdot)$ and to compute z.

Simplified Example: statistical motivation

Bayes formula:

$$p(x|y) = \frac{p(x)p(y|x)}{p(y)}.$$

Simplified Example: statistical motivation

Bayes formula:

$$p(x|y) = \frac{p(x)p(y|x)}{p(y)}.$$

Bayesian statistics: X is hidden, we observe $Y = y$

- $p(x)$ is the *prior* distribution.
- $w(x) = p(y|x)$ is the *likelihood* function.
- $\pi(x) = p(x|y)$ is the *posterior* distribution.

(y is fixed and thus omitted.)

Simplified Example: Importance Sampling

Target distribution:

$$\pi(x) = \frac{p(x)w(x)}{z}.$$

Simplified Example: Importance Sampling

Target distribution:

$$\pi(x) = \frac{p(x)w(x)}{z}.$$

Sampling scheme:

- $N \sim \mathbf{Poiss}(\lambda)$, i.e. $\mathbf{P}(N = n) = \mathbf{e}^{-\lambda}\frac{\lambda^n}{n!}$, $n = 0, 1, 2, \ldots$.
- If $N = 0$ then $\hat{Z} := 0$ end.
- If $N > 0$ then sample $X_1, \ldots, X_N \sim_{\text{iid}} p(\cdot)$,

Simplified Example: Importance Sampling

Target distribution:

$$\pi(x) = \frac{p(x)w(x)}{z}.$$

Sampling scheme:

- $N \sim \text{Poiss}(\lambda)$, i.e. $P(N = n) = e^{-\lambda}\frac{\lambda^n}{n!}$, $n = 0, 1, 2, \ldots$.
- If $N = 0$ then $\hat{Z} := 0$ end.
- If $N > 0$ then sample $X_1, \ldots, X_N \sim_{\text{iid}} p(\cdot)$,

$$\hat{Z} := \frac{1}{\lambda}\sum_{j=1}^{N} w(X_j), \qquad \hat{\pi}(\cdot) = \frac{1}{\lambda}\sum_{j=1}^{N} \delta_{X_j}(\cdot)w(X_j).$$

Simplified Example: Importance Sampling

Target distribution:

$$\pi(x) = \frac{p(x)w(x)}{z}.$$

- $\mathbf{E}w(X_j) = \int p(x)w(x)\mathrm{d}x = z$,
- $\mathbf{E}\left(\sum_{j=1}^{N} w(X_j)\middle|N\right) = Nz$ and $\mathbf{E}N = \lambda$,
- $\mathbf{E}\sum_{j=1}^{N} w(X_j) = \lambda z$.

We thus have $\mathbf{E}\hat{Z} = z$.

Simplified Example: Importance Sampling

Target distribution:

$$\pi(x) = \frac{p(x)w(x)}{z}.$$

- $\mathbf{E}w(X_j) = \int p(x)w(x)\mathrm{d}x = z$,
- $\mathbf{E}\left(\sum_{j=1}^{N} w(X_j) \middle| N\right) = Nz$ and $\mathbf{E}N = \lambda$,
- $\mathbf{E}\sum_{j=1}^{N} w(X_j) = \lambda z$.

We thus have $\mathbf{E}\hat{Z} = z$.

Similarly, $\mathbf{E}\hat{\pi}(B) = \pi(B)$ for every $B \subseteq \mathcal{X}$.

Simplified Example: Importance Sampling

Target distribution:

$$\pi(x) = \frac{p(x)w(x)}{z}.$$

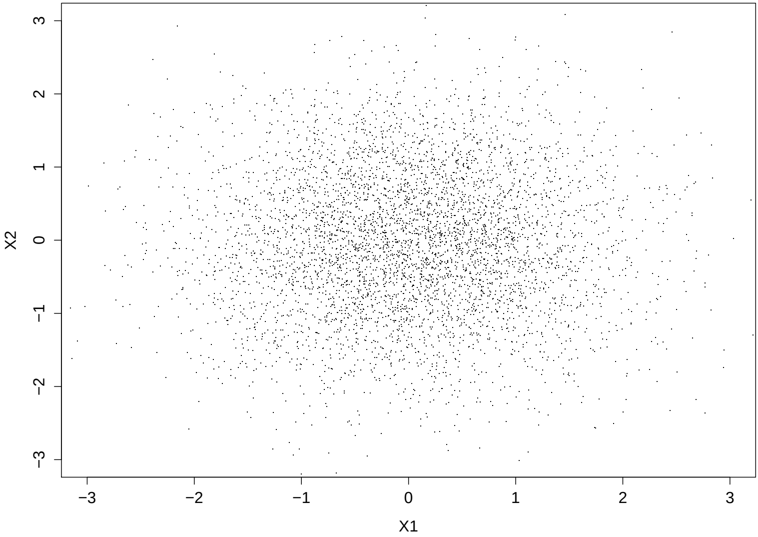
- $\mathbf{E}w(X_j) = \int p(x)w(x)\mathrm{d}x = z$,
- $\mathbf{E}\left(\sum_{j=1}^{N} w(X_j) \middle| N\right) = Nz$ and $\mathbf{E}N = \lambda$,
- $\mathbf{E}\sum_{j=1}^{N} w(X_j) = \lambda z$.

We thus have $\mathbf{E}\hat{Z} = z$.

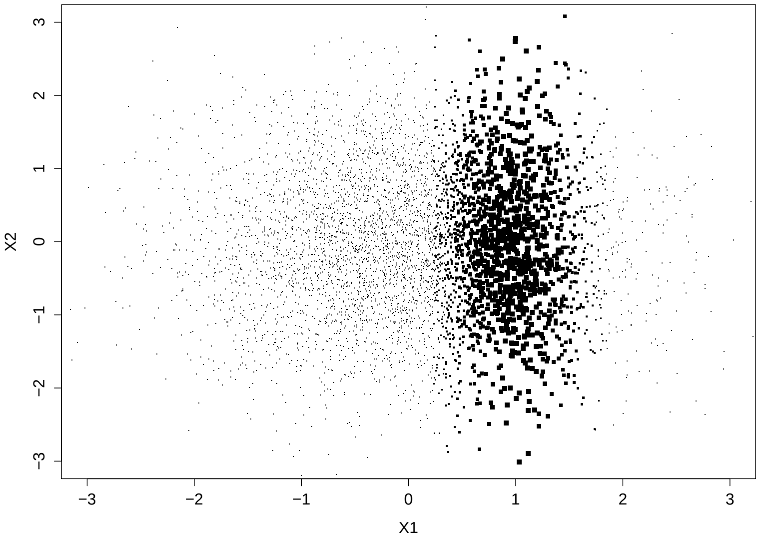
Similarly, $\mathbf{E}\hat{\pi}(B) = \pi(B)$ for every $B \subseteq \mathcal{X}$.

If $\lambda \to \infty$ then $N \to \infty$ and $\hat{\pi}(B) \to \pi(B)$ a.s.

Sample from $p(\cdot)$



Weghted sample from $p(\cdot)$ approximates $\pi(\cdot)$



MCMC: Metropolis-Hastings Algorithm

Target distribution: $\phi(\xi)$ on space Ξ.

We generate Markov chain $\xi^{(0)}, \xi^{(1)}, \ldots, \xi^{(m)}, \ldots \rightarrow \phi(\cdot)$:

MCMC: Metropolis-Hastings Algorithm

Target distribution: $\phi(\xi)$ on space Ξ.

We generate Markov chain $\xi^{(0)}, \xi^{(1)}, \ldots, \xi^{(m)}, \ldots \to \phi(\cdot)$:

- Proposal distribution (transition density): $\psi(\xi, \xi')$.
- Acceptance probability:

$$\alpha(\xi, \xi') = \frac{\phi(\xi')\psi(\xi', \xi)}{\phi(\xi)\psi(\xi, \xi')} \wedge 1.$$

In one step we sample

$$\xi^{(m)} \sim T(\xi^{(m-1)}, \cdot),$$

where, for $\xi' \neq \xi$,

$$T(\xi, \xi') = \psi(\xi, \xi')\alpha(\xi, \xi').$$

and $T(\xi, \{\xi\}) = 1 - \int_{\xi' \neq \xi} T(\xi, \xi') d\xi'$.

MCMC: Metropolis-Hastings Algorithm

One step of MHA from $\xi^{(m-1)}$ to $\xi^{(m)}$:

Sample $\xi' \sim \psi(\xi^{(m-1)}, \cdot)$ { proposal }
Sample $U \sim Unif(0, 1)$
if $U \leqslant \alpha(\xi^{(m-1)}, \xi')$ **then**
  $\xi^{(m)} := \xi'$ { move accepted with probability α }
else
  $\xi^{(m)} := \xi^{(m-1)}$ { move rejected with probability $1 - \alpha$ }
end if

MCMC: Metropolis-Hastings Algorithm

Theorem

Transition kernel of MHA is ϕ-reversible, i.e.

$$\phi(\xi)T(\xi, \xi') = \phi(\xi')T(\xi', \xi)$$

Consequently, ϕ is the stationary (equllibrium) distribution of the chain $\xi^{(m)}$.

Metropolis et al. (1953), Hastings (1970).

Independent Metropolis-Hastings (IMHA)

Proposal distribution $\psi(\xi, \xi') = \psi(\xi')$ is independent of ξ.

Acceptance probability:

$$\alpha(\xi, \xi') = \frac{\phi(\xi')\psi(\xi)}{\phi(\xi)\psi(\xi')} \wedge 1.$$

Simplified Example

Target distribution:

$$\pi(x) = \frac{p(x)w(x)}{z}.$$

Simplified Example

Target distribution:

$$\pi(x) = \frac{p(x)w(x)}{z}.$$

Sampling scheme continued. We draw $(N, X_1, \ldots, X_n, S)$:

- $N \sim \text{Poiss}(\lambda)$, i.e. $P(N = n) = e^{-\lambda}\frac{\lambda^n}{n!}$, $n = 0, 1, 2, \ldots$.
- If $N = 0$ then $\hat{Z} := 0$ end.
- If $N > 0$ then sample $X_1, \ldots, X_N \sim_{\text{iid}} p(\cdot)$,

$$\hat{Z} := \frac{1}{\lambda}\sum_{j=1}^{N} w(X_j).$$

Choose $S \in \{1, \ldots, N\}$ at random:

$$P(S = s | N, X_1, \ldots, X_N) = \frac{w(X_s)}{\sum_{j=1}^{N} w(X_j)}.$$

Simplified Example

ψ – joint probability distribution of all random variables:

$$\psi(n, x_1, \ldots, x_n, s) = e^{-\lambda} \frac{\lambda^n}{n!} \prod_{j=1}^{n} p(x_j) \frac{w(x_s)}{\sum_{j=1}^{n} w(x_j)}.$$

for $n > 0$ and $\psi(0) = e^{-\lambda}$.

Simplified Example

ψ – joint probability distribution of all random variables:

$$\psi(n, x_1, \ldots, x_n, s) = e^{-\lambda} \frac{\lambda^n}{n!} \prod_{j=1}^{n} p(x_j) \frac{w(x_s)}{\sum_{j=1}^{n} w(x_j)}.$$

for $n > 0$ and $\psi(0) = e^{-\lambda}$.

Lemma

$$\psi(n, x_1, \ldots, x_n, s) = \frac{z}{\hat{z}} \phi(n, x_1, \ldots, x_n, s),$$

where ϕ is a probability distribution such that $\phi(0) = 0$ and

$$\phi(x_s) = \pi(x_s).$$

Marginal of ϕ is the target π!

Simplified Example

Proof of Lemma:

$$
\begin{aligned}
\psi(n, x_1, \ldots, x_n, s) &= e^{-\lambda}\frac{\lambda^n}{n!}\prod_{j=1}^{n} p(x_j)\frac{w(x_s)}{\sum_{j=1}^{n} w(x_j)} \\
&= \frac{1}{z}p(x_s)w(x_s)\,\frac{z}{\hat{z}}\,\frac{1}{n}e^{-\lambda}\frac{\lambda^{n-1}}{(n-1)!}\prod_{j\neq s} p(x_j) \\
&= \underbrace{\pi(x_s)}_{target}\frac{z}{\hat{z}}\cdot\frac{1}{n}\psi(n-1, x_{-s}).
\end{aligned}
$$

□

Simplified Example: IMHA

Independent Metropolis-Hastings chain on the **extended space** – space of configurations $\xi = (\boldsymbol{n}, \boldsymbol{x}, \boldsymbol{s})$:

- If the current state is $\xi^{(m-1)} = (\boldsymbol{N}, \boldsymbol{X}, \boldsymbol{S})$ then
- Draw a proposal: $(\boldsymbol{N}', \boldsymbol{X}', \boldsymbol{S}') \sim \psi$ (sampling scheme as described). Compute

$$\alpha := \frac{\phi(\boldsymbol{N}', \boldsymbol{X}', \boldsymbol{S}')\psi(\boldsymbol{N}, \boldsymbol{X}, \boldsymbol{S})}{\phi(\boldsymbol{N}, \boldsymbol{X}, \boldsymbol{S})\psi(\boldsymbol{N}', \boldsymbol{X}', \boldsymbol{S}')} \wedge 1 = \frac{\hat{Z}'}{\hat{Z}} \wedge 1.$$

  - with probability α accept: $\xi^{(m)} := (\boldsymbol{N}', \boldsymbol{X}', \boldsymbol{S}')$;
  - with probability $1 - \alpha$ reject: $\xi^{(m)} := (\boldsymbol{N}, \boldsymbol{X}, \boldsymbol{S})$.

Simplified Example: IMHA

Independent Metropolis-Hastings chain on the **extended space** – space of configurations $\xi = (\boldsymbol{n}, \boldsymbol{x}, \boldsymbol{s})$:

- If the current state is $\xi^{(m-1)} = (\boldsymbol{N}, \boldsymbol{X}, \boldsymbol{S})$ then
- Draw a proposal: $(\boldsymbol{N}', \boldsymbol{X}', \boldsymbol{S}') \sim \psi$ (sampling scheme as described). Compute

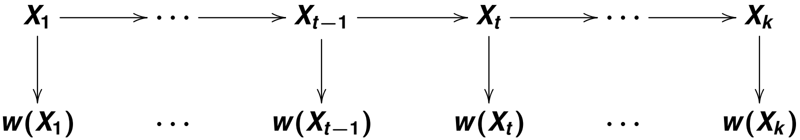
$$\alpha := \frac{\phi(\boldsymbol{N}', \boldsymbol{X}', \boldsymbol{S}')\psi(\boldsymbol{N}, \boldsymbol{X}, \boldsymbol{S})}{\phi(\boldsymbol{N}, \boldsymbol{X}, \boldsymbol{S})\psi(\boldsymbol{N}', \boldsymbol{X}', \boldsymbol{S}')} \wedge 1 = \frac{\hat{\boldsymbol{Z}}'}{\hat{\boldsymbol{Z}}} \wedge 1.$$

  - with probability α accept: $\xi^{(m)} := (\boldsymbol{N}', \boldsymbol{X}', \boldsymbol{S}')$;
  - with probability $1 - \alpha$ reject: $\xi^{(m)} := (\boldsymbol{N}, \boldsymbol{X}, \boldsymbol{S})$.

The chain preserves $\phi(\boldsymbol{n}, \boldsymbol{x}, \boldsymbol{s})$ thus marginally $\pi(\boldsymbol{x}_{\boldsymbol{s}})$. It converges to the target distribution.

Extended Proposal: ψ, *Extended Target*: ϕ.

Hidden Markov Model



- $X = X_{1:k} = (X_1, \ldots, X_k)$ is a hidden Markov chain with transition kernel $p(x_{t-1}, x_t) = p(x_t | x_{t-1})$.
- Likelihood weights : $w(x_t) = p(y_t | x_t)$, where y_t is observed.

Target (posterior) distribution:

$$\pi(x_{1:k}) = \frac{1}{z} \prod_{t=1}^{k} p(x_{t-1}, x_t) w(x_t).$$

Algorytm PTPF (Poisson Tree Particle Filter)

Directed tree with “marked” nodes:

$$(\mathcal{V}, \mathcal{E}, \mathbf{X} = \{X_v, v \in \mathcal{V}\}, S).$$

Notations:

- $\mathbf{0}$ – fictitious root,
- $\mathbf{ch}(v)$ – children of node $v \in \mathcal{V}$,
- $\mathbf{an}(v)$ – ancestors of $v \in \mathcal{V}$ (with v, without $\mathbf{0}$),
- $\mathcal{V}_t$ – tth generation ($t = 1, \ldots, k$),
- S – selected final node $\in \mathcal{V}_k$.

Algorithm PTPF (Poisson Tree Particle Filter)

Sampling scheme: For $t = 0, 1, \ldots, k$, for every node $v \in \mathcal{V}_t$

- Choose Λ_v depending on history,
- $N_v \sim \mathbf{Poiss}(\Lambda_v w(X_v))$,
- Create set $\mathbf{ch}(v)$ of cardinality N_v,
- For every $u \in \mathbf{ch}(v)$ sample $X_u \sim p(X_v, \cdot)$ propagate,
- Compute $W_u := w(X_u)$ weigh

until $t = k$ or $\sum_{v \in \mathcal{V}_t} N_t = 0$;

Algorithm PTPF (Poisson Tree Particle Filter)

Sampling scheme: For $t = 0, 1, \ldots, k$, for every node $v \in \mathcal{V}_t$

- Choose Λ_v depending on history,
- $N_v \sim \mathbf{Poiss}(\Lambda_v w(X_v))$,
- Create set $\mathbf{ch}(v)$ of cardinality N_v,
- For every $u \in \mathbf{ch}(v)$ sample $X_u \sim p(X_v, \cdot)$ propagate,
- Compute $W_u := w(X_u)$ weigh

until $t = k$ or $\sum_{v \in \mathcal{V}_t} N_t = 0$;

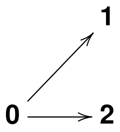
$\hat{Z} := \sum_{u \in \mathcal{V}_k} w(X_u)/C_u$, where $C_u = \lambda_0 \prod_{i \in \mathbf{an}(u) - u} \Lambda_v$,

Select $S \in \mathcal{V}_k$: $\mathbf{P}(S = s) \propto w(X_s)/C_s$.

Example: *extended proposal*

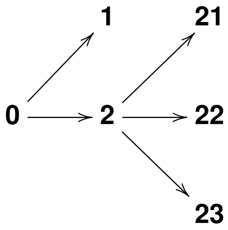
0

Example: *extended proposal*



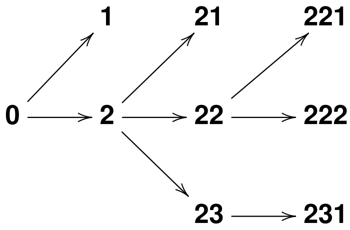
$$\psi = \exp\left[-\lambda_0\right](\lambda_0)^2 p(x_0, x_1)p(x_0, x_2)$$

Example: *extended proposal*



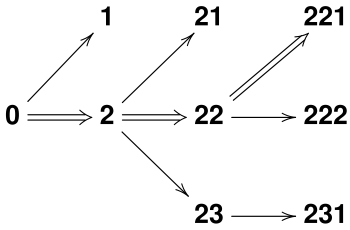
$$\psi = \exp[-\lambda_0](\lambda_0)^2 p(x_0, x_1) p(x_0, x_2)$$
$$\times \exp[-\lambda_1 w_1]$$
$$\times \exp[-\lambda_2 w_2](\lambda_2 w_2)^3 p(x_2, x_{21}) p(x_2, x_{22}) p(x_2, x_{23})$$

Example: *extended proposal*



$$
\begin{aligned}
\psi = {} & \exp\left[-\lambda_0\right](\lambda_0)^2 p(x_0, x_1) p(x_0, x_2) \\
& \times \exp\left[-\lambda_1 w_1\right] \\
& \times \exp\left[-\lambda_2 w_2\right](\lambda_2 w_2)^3 p(x_2, x_{21}) p(x_2, x_{22}) p(x_2, x_{23}) \\
& \times \exp\left[-\lambda_{21} w_{21}\right] \\
& \times \exp\left[-\lambda_{22} w_{22}\right](\lambda_{22} w_{22})^2 p(x_{22}, x_{221}) p(x_{22}, x_{222}) \\
& \times \exp\left[-\lambda_{23} w_{23}\right](\lambda_{23} w_{23})^1 p(x_{23}, x_{231})
\end{aligned}
$$

Example: *extended proposal*



$$
\begin{aligned}
\psi = & \exp\left[-\lambda_0\right](\lambda_0)^2 p(x_0, x_1) p(x_0, x_2) \\
& \times \exp\left[-\lambda_1 w_1\right] \\
& \times \exp\left[-\lambda_2 w_2\right](\lambda_2 w_2)^3 p(x_2, x_{21}) p(x_2, x_{22}) p(x_2, x_{23}) \\
& \times \exp\left[-\lambda_{21} w_{21}\right] \\
& \times \exp\left[-\lambda_{22} w_{22}\right](\lambda_{22} w_{22})^2 p(x_{22}, x_{221}) p(x_{22}, x_{222}) \\
& \times \exp\left[-\lambda_{23} w_{23}\right](\lambda_{23} w_{23})^1 p(x_{23}, x_{231}) \\
& \qquad \times \frac{1}{\hat{z}} \frac{w_{221}}{\lambda_0 \lambda_2 \lambda_{22}}; \qquad \hat{z} = \frac{w_{221}}{\lambda_0 \lambda_2 \lambda_{22}} + \frac{w_{222}}{\lambda_0 \lambda_2 \lambda_{22}} + \frac{w_{231}}{\lambda_0 \lambda_2 \lambda_{23}}
\end{aligned}
$$

Extended proposal and *extended target*

Extended proposal ψ is the joint probability distribution of all the variables produced by PTPF.

Lemma

If $\mathcal{V}_k \neq \emptyset$ then

$$\psi(\mathcal{V}, \mathcal{E}, \mathbf{x}, \boldsymbol{s}) = \phi(\mathcal{V}, \mathcal{E}, \mathbf{x}, \boldsymbol{s}) \frac{\boldsymbol{z}}{\hat{\boldsymbol{z}}},$$

*where ϕ (**extended target**) is such that the marginal of $\boldsymbol{x}_{\mathrm{an}(\boldsymbol{s})}$ is the target: $\phi(\boldsymbol{x}_{\mathrm{an}(\boldsymbol{s})}) = \pi(\boldsymbol{x}_{\mathrm{an}(\boldsymbol{s})})$ and $\phi(\mathbf{0}) = \mathbf{0}$.*

Poisson Tree IMHA

Independent Metropolis-Hastings chain on the **extended space** – space of **trees** $\xi = (\mathcal{V}, \mathcal{E}, \mathbf{x} = \{\mathbf{x}_v, v \in \mathcal{V}\}, s)$:

- If the current state is $\xi^{(m-1)}$ then
- Draw a proposal: run PTPF to obtain $\xi' \sim \psi$. Compute $\hat{Z}'$.

$$\alpha := \frac{\phi(\xi')\psi(\xi)}{\phi(\xi)\psi(\xi')} \wedge 1 = \frac{\hat{Z}'}{\hat{Z}} \wedge 1.$$

  - with probability α accept: $\xi^{(m)} := \xi'$;
  - with probability $1 - \alpha$ reject: $\xi^{(m)} := \xi^{(m-1)}$.

Poisson Tree IMHA

Independent Metropolis-Hastings chain on the **extended space** – space of **trees** $\xi = (\mathcal{V}, \mathcal{E}, \mathbf{x} = \{\mathbf{x}_v, v \in \mathcal{V}\}, s)$:

- If the current state is $\xi^{(m-1)}$ then
- Draw a proposal: run PTPF to obtain $\xi' \sim \psi$. Compute $\hat{Z}'$.

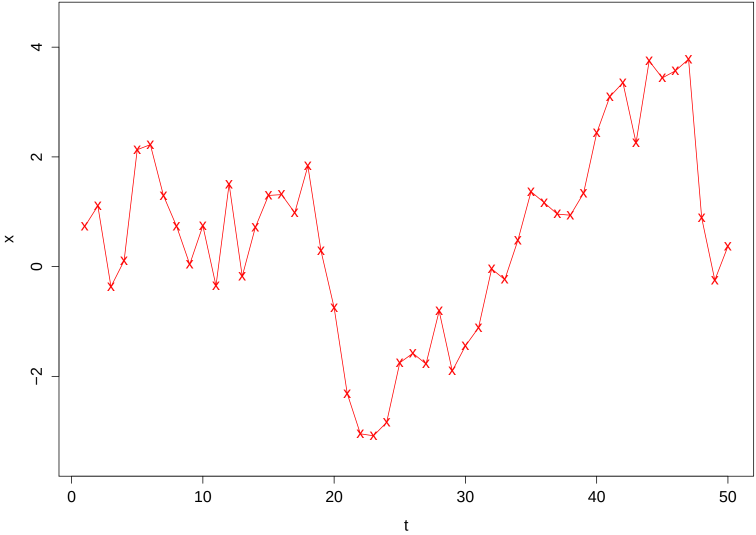
$$\alpha := \frac{\phi(\xi')\psi(\xi)}{\phi(\xi)\psi(\xi')} \wedge 1 = \frac{\hat{Z}'}{\hat{Z}} \wedge 1.$$

  - with probability α accept: $\xi^{(m)} := \xi'$;
  - with probability $1 - \alpha$ reject: $\xi^{(m)} := \xi^{(m-1)}$.

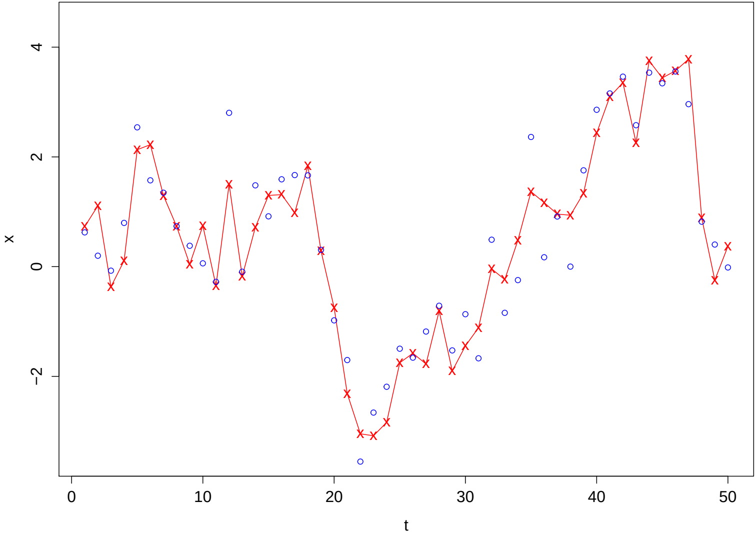
The chain preserves $\phi(\xi)$ thus marginally $\pi(\mathbf{x}_{\mathrm{an}(s)})$. It converges to the target distribution.

Extended Proposal: ψ, *Extended Target*: ϕ.

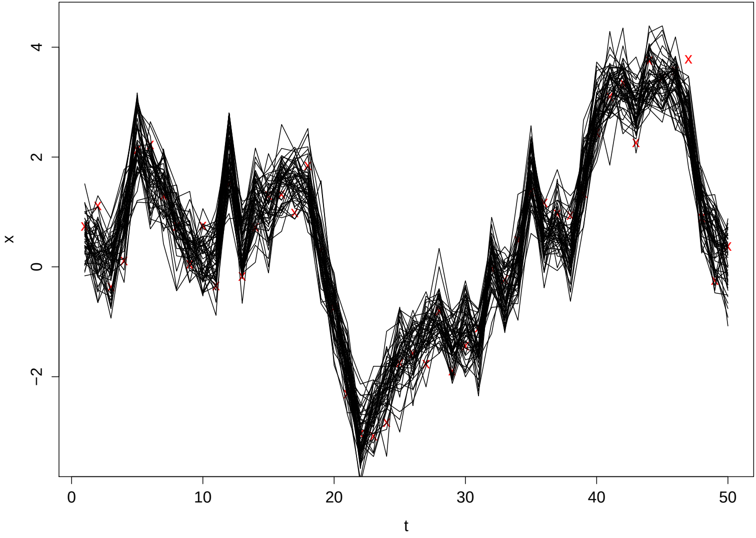
Hidden process



Hidden process and observations



Trajectories sampled from the posterior via PTMC



What is new in PTPF?

Parallelization of computations:

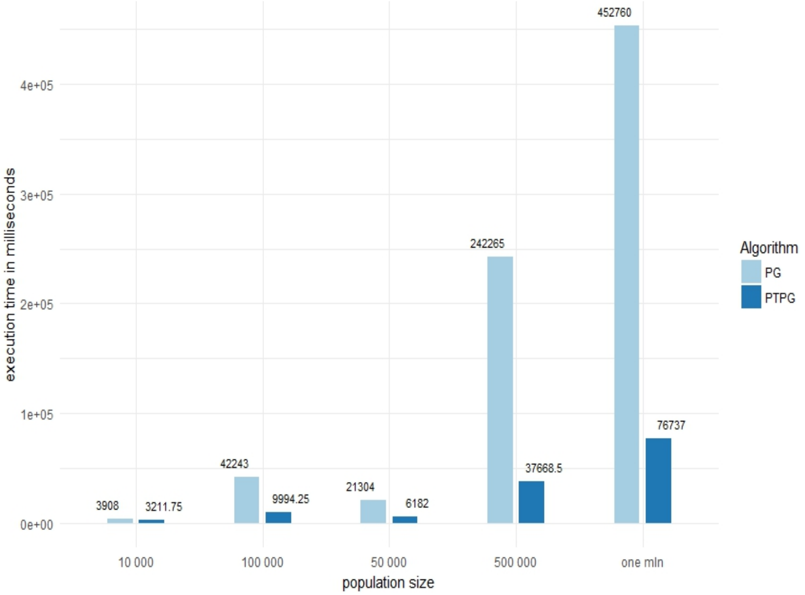
- If $\boldsymbol{\Lambda_v}$ depends only on $\mathbf{an}(\boldsymbol{v}) - \boldsymbol{v}$ then the branches of the tree evolve **completely independently.**

What is new in PTPF?

Parallelization of computations:

- If $\boldsymbol{\Lambda_v}$ depends only on $\mathbf{an}(\boldsymbol{v}) - \boldsymbol{v}$ then the branches of the tree evolve **completely independently.**
- If the branches evolve **partly independently**, we can control their number.

Parallel computations are more efficient



What is new in PTPF?

Continuous time models:

- PTMC can be directly applied to hidden piecewise deterministic (quasi-) Markov processes.

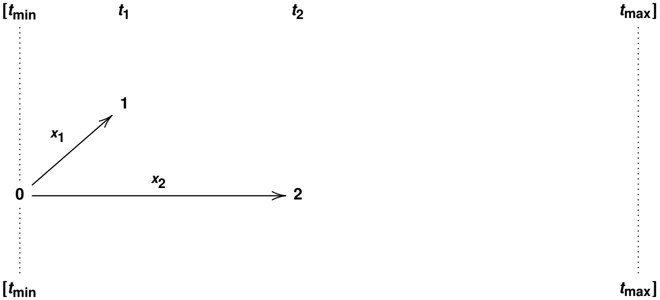
What is new in PTPF?

Continuous time models:

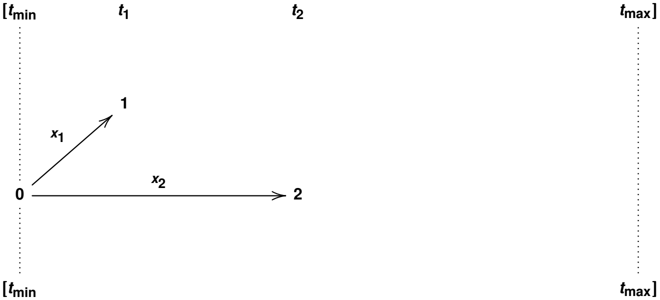
- PTMC can be directly applied to hidden piecewise deterministic (quasi-) Markov processes.
- If the branches evolve **partly independently**, we can control their number.

A tree with space-time nodes

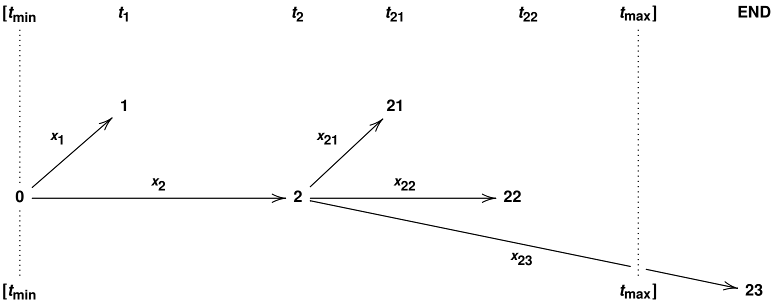
A tree with *space-time* nodes



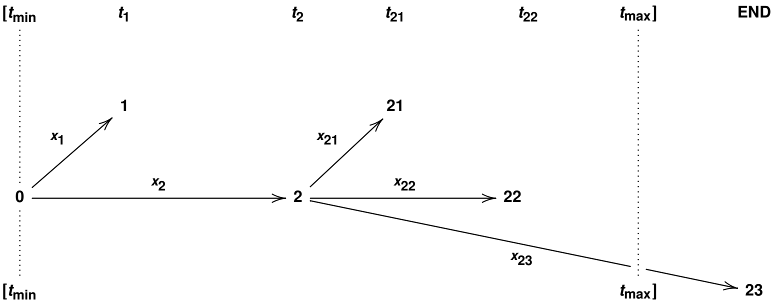
A tree with *space-time* nodes



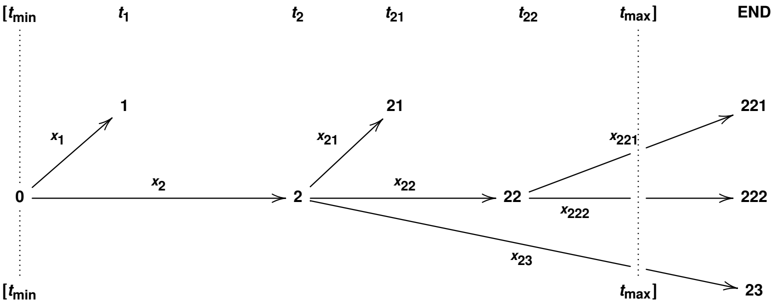
A tree with *space-time* nodes



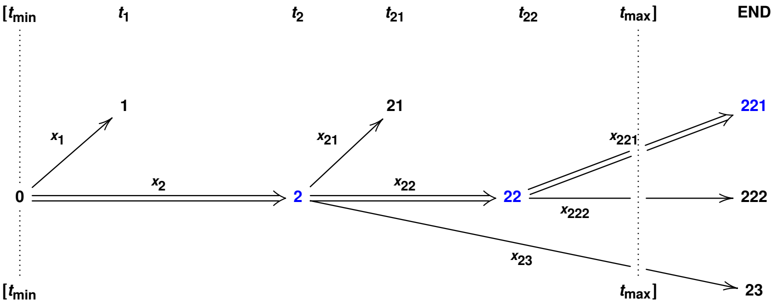
A tree with *space-time* nodes



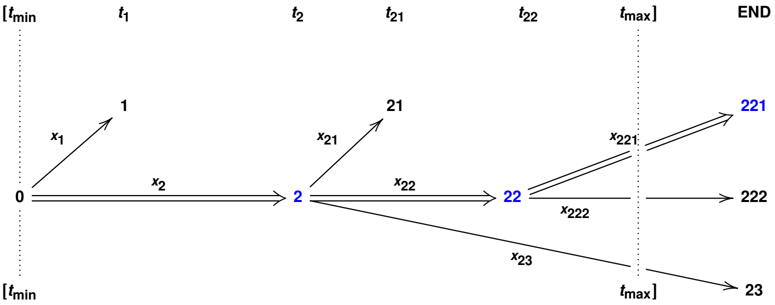
A tree with *space-time* nodes



A tree with *space-time* nodes



A tree with *space-time* nodes



References

- C. Andrieu, A. Doucet, R. Holenstein: Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society B, 2010*
- F. Lindsten, M.I. Jordan, T.B. Schön: Particle Gibbs with Ancestor Sampling, *Journal of Machine Learning Research, 2014*
- F. Lindsten, R. Douc, E. Moulines: Uniform ergodicity of the Particle Gibbs sampler, *Scandinavian Journal of Statistics, 2015*
- T. Cąkała, B. Miasojedow, W. Niemiro: Particle MCMC algorithms with Poisson resampling: parallelization and continuous time models. `arXiv:1707.01660v2`)