# Perfect simulation algorithm of a trajectory under a Feynman-Kac law

## University of Warwick, Recent Advances in Sequential Monte Carlo - Sep 19-21, 2012

C. Andrieu, A. Doucet, S. Rubenthaler

University of Bristol, University of Oxford, Université de Nice-Sophia Antipolis

# Notations

- $X_1, X_2, \dots$ is a Markov chain in $E$ with initial law $M_1$ and transition $M$ (say $E = \mathbb{R}^d$ or $\mathbb{Z}^d$)
- $G_1, G_2, \dots : E \to \mathbb{R}_+$ are potentials
- total time : $P$

One is interested in the law :

$$\pi(f) = \frac{\mathbb{E}(f(X_1, \dots, X_P) \prod_{i=1}^{P-1} G_i(X_i))}{\mathbb{E}(\prod_{i=1}^{P-1} G_i(X_i))}.$$

## Simple branching system

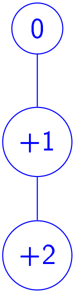
- Start with $N_1$ particles.
- The particle $X_n^i$ ($i$-th particle at time $n$) has $A_{n+1}^i$ offsprings with law $\mathbb{P}(A_{n+1}^i = j) = f_{n+1}(G_n(X_n^i), j)$ (independant of other particles).
- Total number of particles : $N_{n+1} = \sum_{i=1}^{N_n} A_{n+1}^i$.

Density :

$$q_0(N_1, \dots, N_P, (A_n^i), (X_n^i)) = \prod_{i=1}^{N_1} M_1(X_1^i) \prod_{n=2}^{P} \left( \prod_{i=1}^{N_{n-1}} f_n(G_{n-1}(X_{n-1}^i), A_n^i) \prod_{j \in \dots} M(X_{n-1}^i, X_n^j) \right)$$

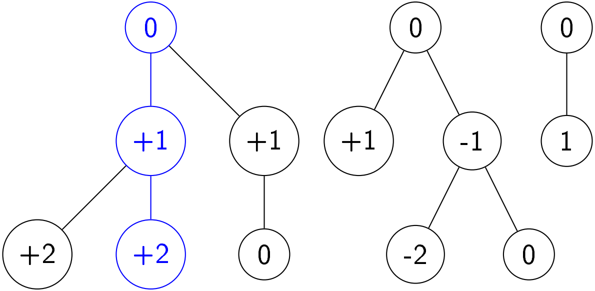
# Extension

Take a trajectory and draw a branching system conditionned to contain this trajectory.



# Extension

Take a trajectory and draw a branching system conditionned to contain this trajectory.



# Extension

When a particle is blue at position $x$ at time $n-1$, the number of children is chosen with law :

$$\mathbb{P}(j \text{ offsprings}) = \widehat{f}_n(G_{n-1}(x), j) = \frac{f_n(G_{n-1}(x), j)}{1 - f_n(G_{n-1}(x), 0)}.$$

# Extension

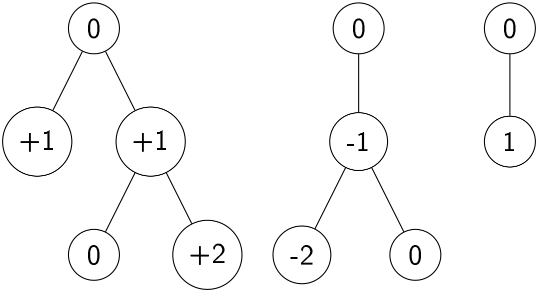
When a particle is blue at position $x$ at time $n-1$, the number of children is chosen with law :

$$\mathbb{P}(j \text{ offsprings}) = \widehat{f}_n(G_{n-1}(x), j) = \frac{f_n(G_{n-1}(x), j)}{1 - f_n(G_{n-1}(x), 0)}.$$

We choose $f_n$ such that $\frac{\widehat{f}_n(g,j)}{f_n(g,j)} = \frac{\|G_n\|_\infty}{g}$ $(\forall n, j, g)$ (take $f_n(g, 0) = 1 - \frac{g}{\|G_n\|_\infty}$, $f_n(g, j) = \frac{g}{k_n \|G_n\|_\infty}$, $1 \le j \le k_n$).

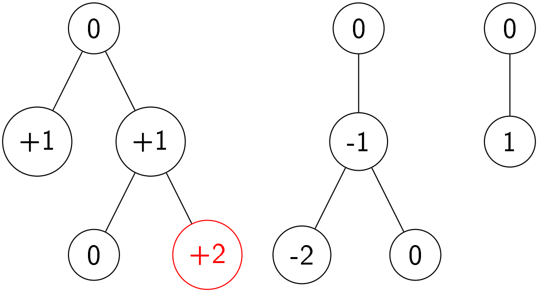
# Proposal

Take a branching system like above, select a particle at time $P$ and its ancestral line. We get some density $q$ on the space of (size of each generation)×(numbers of offsprings)×(positions)×(special trajectory).



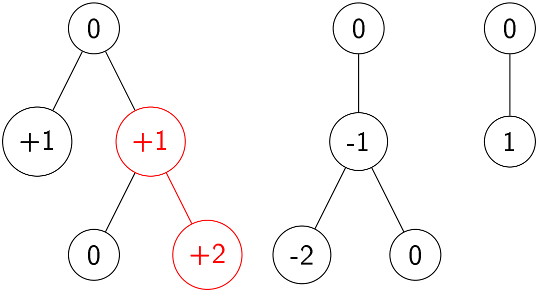
# Proposal

Take a branching system like above, select a particle at time $P$ and its ancestral line. We get some density $q$ on the space of (size of each generation)×(numbers of offsprings)×(positions)×(special trajectory).



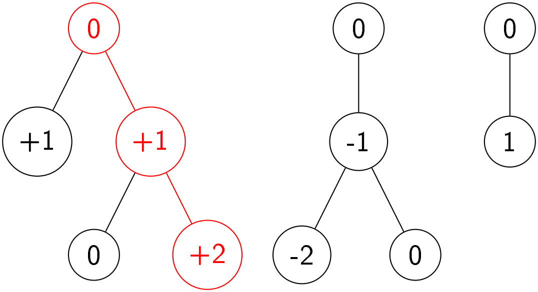
# Proposal

Take a branching system like above, select a particle at time $P$ and its ancestral line. We get some density $q$ on the space of (size of each generation)$\times$(numbers of offsprings)$\times$(positions)$\times$(special trajectory).



# Proposal

Take a branching system like above, select a particle at time $P$ and its ancestral line. We get some density $q$ on the space of (size of each generation)$\times$(numbers of offsprings)$\times$(positions)$\times$(special trajectory).



# Acception/rejection

Target law : trajectory with the law $\pi$, to which we add a (conditionned) branching system. We have a law $\widehat{\pi}$ on "forests".

$$\frac{\widehat{\pi}(\dots)}{q(\dots)} = \frac{N_P \prod_{i=1}^{P-1} \|G_i\|_\infty}{N_1 Z},$$

with $Z := \mathbb{E}(\prod_{n=1}^{P-1} G_n(X_n))$ (partition function).

# Markov chain

- start with trajectory, extend it into a "forest"

# Markov chain

- start with trajectory, extend it into a "forest"
- when at point $\mathcal{X}$ (in "forests"): propose $\overline{\mathcal{X}}$,

# Markov chain

- start with trajectory, extend it into a "forest"
- when at point $\mathcal{X}$ (in "forests"): propose $\overline{\mathcal{X}}$,
- accept with probability $\frac{\widehat{\pi}(\overline{\mathcal{X}})q(\mathcal{X})}{\widehat{\pi}(\mathcal{X})q(\overline{\mathcal{X}})} = \frac{\overline{N}_P}{N_P}$,

# Markov chain

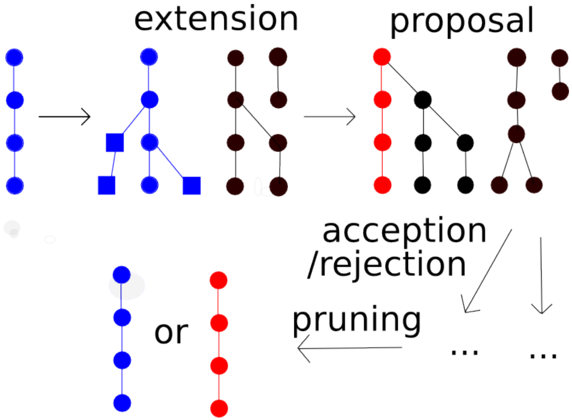
- start with trajectory, extend it into a "forest"
- when at point $\mathcal{X}$ (in "forests"): propose $\overline{\mathcal{X}}$,
- accept with probability $\frac{\widehat{\pi}(\overline{\mathcal{X}})q(\mathcal{X})}{\widehat{\pi}(\mathcal{X})q(\overline{\mathcal{X}})} = \frac{\overline{N}_P}{N_P}$,
- then prune the forest to have only the colored trajectory.

# Markov chain

- start with trajectory, extend it into a "forest"
- when at point $\mathcal{X}$ (in "forests"): propose $\overline{\mathcal{X}}$,
- accept with probability $\frac{\widehat{\pi}(\overline{\mathcal{X}})q(\mathcal{X})}{\widehat{\pi}(\mathcal{X})q(\overline{\mathcal{X}})} = \frac{\overline{N}_P}{N_P}$,
- then prune the forest to have only the colored trajectory.

We have here a Markov process on the trajectory space whose invariant law is $\pi$.

# Markov chain



# Coupling from the past in a nutshell

Transition of a Markov chain $(Z_k)$ expressed with i.i.d. variables $(U_k)$:

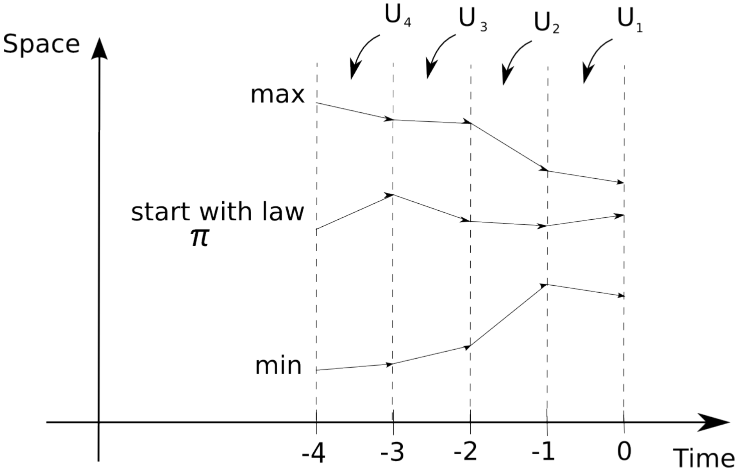
$$Z_{k+1} = h(Z_k, U_{k+1}).$$

Suppose $(Z_k)$ has invariant law $\pi$. For a starting point $z$ and $n \geq 0$, set

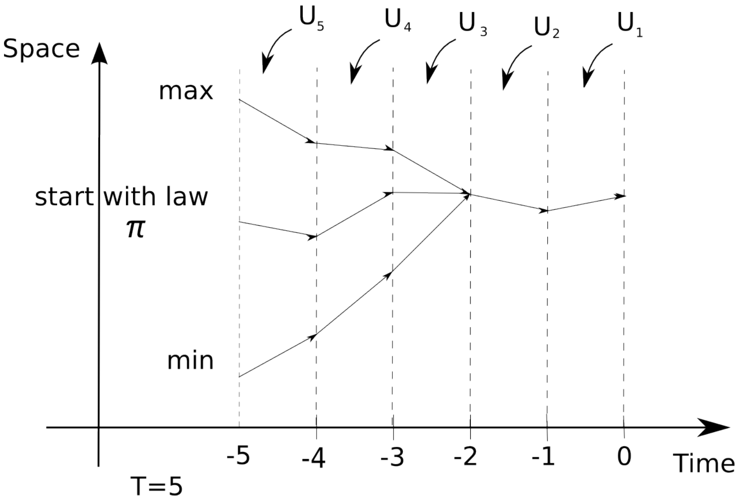
$$Z^z_{-n} = z\,,\; Z^z_{-n+1} = h(Z^z_{-n}, U_n)\,, \dots,\; Z^z_0 = h(Z^z_{-1}, U_1).$$

If $T$ such that $Z^z_0 = Z^{z'}_0$, $\forall z, z'$: $Z^z_{-T} = z$, $Z^{z'}_{-T} = z'$, then $Z^z_0 \sim \pi$.

# Coupling from the past algorithm

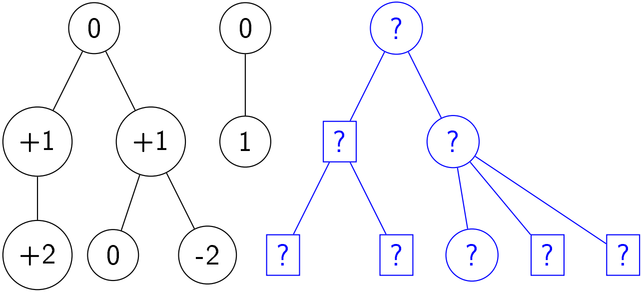


# Coupling from the past algorithm



# Detection of a coupling time

Look for a time such that the red proposal is accepted for all possible blue trajectories.




Bound the number of squares at the bottom and you bound the acceptation ratio.

# Directed polymers in $\mathbb{Z}$

Draw $U(i,j)$ i.i.d. of Bernoulli law ($i \in \mathbb{N}$, $j \in \mathbb{Z}$). Take $(X_n)$ the simple random walk in $\mathbb{Z}$ with $X_0 = 0$. Set $G_i(j) = \exp(-\beta U(i,j))$. To draw a trajectory of length $n$, the cost is $O(n^2)$.

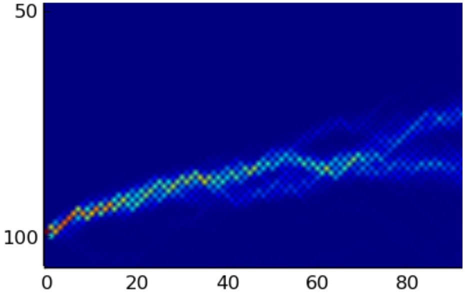



Figure: 500 trajectories

# Directed polymers in $\mathbb{Z}$

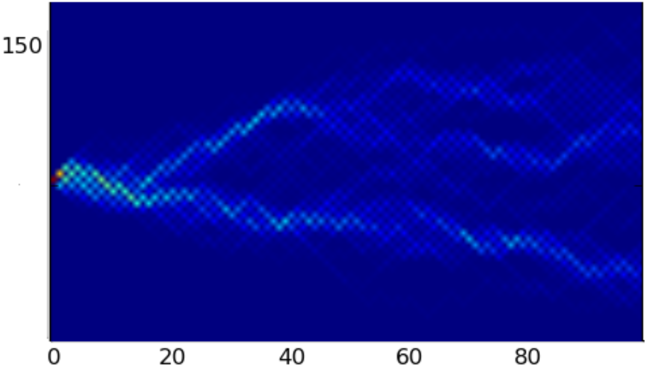



Figure: 100 trajectories

# Radar detection

Transition (in $\mathbb{R}$) : $M(x,dy) = \frac{1}{\sqrt{2\pi b^2}} \exp\left(-\frac{(y-ax)^2}{2b^2}\right)$,
$G_n(x) = \frac{1}{\sqrt{2\pi c^2}} \exp\left(-\frac{(x-Y_n)^2}{2c^2}\right)$ , where $(X_n)$ has transition $M$ and $Y_n = X_n + c\epsilon_n$ $(\epsilon_n \sim \mathcal{N}(0,1))$.
You can bound the number of offspring of any trajectory by discretizing the space (works for $a \in [-1,1]$).

– : X
– : Y
– : perfect simulation