On Sequential Monte Carlo (SMC) strategies for Target Distributions

M. Rousset [1,2]
F. Cérou, A. Guyader, B. Delyon, T. Lelièvre, G. Stoltz, C.E. Bréhier, L. Goudenège, P. Héas. (PhDs: F. Ernoult, K. Tit).

[1]Inria Rennes Bretagne Atlantique

[2]IRMAR, Université de Rennes 1

2021

Aim of the talk

Aim of the talk

- **'Target probability distribution'**: defined as a density w.r.t to a simulable distribution, density given up to a normalizing constant. E.g.: posterior or conditional distribution, Gibbs probability.

Aim of the talk

- **'Target probability distribution'**: defined as a density w.r.t to a simulable distribution, density given up to a normalizing constant. E.g.: posterior or conditional distribution, Gibbs probability.
- **SMC = particle methods= Importance splitting**. As "opposed" to MCMC methods. Start with a sample of N replicas ('particles'). Algorithms output: sample of N particles (approx. indep.) with distribution the 'target' + estimator of normalisation.

Aim of the talk

- **'Target probability distribution'**: defined as a density w.r.t to a simulable distribution, density given up to a normalizing constant. E.g.: posterior or conditional distribution, Gibbs probability.
- **SMC = particle methods= Importance splitting**. As "opposed" to MCMC methods. Start with a sample of N replicas ('particles'). Algorithms output: sample of N particles (approx. indep.) with distribution the 'target' + estimator of normalisation.
- **Aim of the talk:** Overview on variants and adaptivity + app. to neutrons (cf. talk of Lelièvre and Del Moral).

E.g.: Rare event problem

E.g.: Rare event problem

- $\pi(dx)$ a reference probability on S that can be simulated.

E.g.: Rare event problem

- $\pi(dx)$ a reference probability on S that can be simulated.
- $\mathrm{score} : \mathbb{R}^d \to \mathbb{R}$ a given computable function.

E.g.: Rare event problem

- $\pi(dx)$ a reference probability on S that can be simulated.
- $\mathrm{score} : \mathbb{R}^d \to \mathbb{R}$ a given computable function.
- Assume $\pi(\{\mathrm{score} > 0\}) = 1$. Problem for given s:

$$\begin{cases} \text{Estimate} \quad p_s := \pi(\{\mathrm{score} > s\}) \ll 1 \\ \text{Simulate according to } \underline{\text{'target'}} \quad \eta_s(dx) := \pi(dx | \mathrm{score}(x) > s). \end{cases}$$

Idea (For high dimensions / low temperature)

Estimate/Simulate **"smoothly" and sequentially** *the path*

$$s \mapsto (p_s, \eta_s), \quad s \in [0, 1].$$

Generalization

Generalization

- $\frac{1}{z_0}\mathrm{e}^{-V_0(0)}\pi(dx)$ a reference probability on $S = \mathbb{R}^d$ that can be simulated, e.g. $z_0 = 1$.

Generalization

- $\frac{1}{z_0}\mathrm{e}^{-V_0(0)}\pi(dx)$ a reference probability on $S = \mathbb{R}^d$ that can be simulated, e.g. $z_0 = 1$.
- $(s, x) \mapsto V_s(x) : \mathbb{R} \times \mathbb{R}^d \times \to \mathbb{R}$ a given computable function (called potential). (Optional: $\nabla_x V_s(x)$ is available).

Generalization

- $\frac{1}{z_0}\mathrm{e}^{-V_0(0)}\pi(dx)$ a reference probability on $S = \mathbb{R}^d$ that can be simulated, e.g. $z_0 = 1$.
- $(s,x) \mapsto V_s(x) : \mathbb{R} \times \mathbb{R}^d \times \to \mathbb{R}$ a given computable function (called potential). (Optional: $\nabla_x V_s(x)$ is available).
- Problem, for $s := 1$:

$$\begin{cases} \text{Estimate the normalization:} \quad z_s := \pi(\mathrm{e}^{-V_s()}) \\ \text{Simulate according to 'target':} \quad \eta_s(dx) := \frac{1}{z_s}\mathrm{e}^{-V_s(x)}\pi(dx). \end{cases}$$

Generalization

- $\frac{1}{z_0}\mathrm{e}^{-V_0(0)}\pi(dx)$ a reference probability on $S = \mathbb{R}^d$ that can be simulated, e.g. $z_0 = 1$.
- $(s, x) \mapsto V_s(x) : \mathbb{R} \times \mathbb{R}^d \times \to \mathbb{R}$ a given computable function (called potential). (Optional: $\nabla_x V_s(x)$ is available).
- Problem, for $s := 1$:

$$\begin{cases} \text{Estimate the normalization:} & z_s := \pi(\mathrm{e}^{-V_s()}) \\ \text{Simulate according to 'target':} & \eta_s(dx) := \frac{1}{z_s}\mathrm{e}^{-V_s(x)}\pi(dx). \end{cases}$$

- Previous rare event model is particular case for:

$$V_s(x) = \begin{cases} +\infty & \text{if } \mathrm{score}(x) \leqslant s \\ 0 & \text{if } \mathrm{score}(x) > s \end{cases}$$

Manifold Generalization[1]

[1] Lelièvre-Stoltz-Rousset, *Langevin dynamics with constraints and computation of free energy differences*, 2012

Manifold Generalization[1]

- $\frac{1}{z_0}\mathrm{e}^{-V(x,0)}\pi_0(dx)$ a target probability on $S = \mathbb{R}^d$ that can be simulated. $z_0 = 1$.

[1] Lelièvre-Stoltz-Rousset, *Langevin dynamics with constraints and computation of free energy differences*, 2012

Manifold Generalization[1]

- $\frac{1}{z_0}\mathrm{e}^{-V(x,0)}\pi_0(dx)$ a target probability on $S = \mathbb{R}^d$ that can be simulated. $z_0 = 1$.
- Target : $\mathrm{e}^{-V_s} d\pi_s / z_s$.

[1] Lelièvre-Stoltz-Rousset, *Langevin dynamics with constraints and computation of free energy differences*, 2012

Manifold Generalization[1]

- $\frac{1}{z_0} \mathrm{e}^{-V(x,0)} \pi_0(dx)$ a target probability on $S = \mathbb{R}^d$ that can be simulated. $z_0 = 1$.
- Target : $\mathrm{e}^{-V_s} d\pi_s / z_s$.
- $s \mapsto \pi_s$ a path of mutually singular non-negative reference measures and a family of computable maps $i_{s,s'} : \mathbb{R}^d \to \mathbb{R}^d$ with $s, s' \in \mathbb{R}$ such that:

$$\pi_{s'} = i_{s,s'}[\pi_s] \quad \text{(push-forward)}$$

Example

$\pi_s := 2d' < 2d$-dimensional phase-space volume of a parametric family of co-tangent spaces $s \mapsto T^*\Sigma_s \subset \mathbb{R}^{2d}$. $i_{s,s'}$ is a simulable symplectic projection.

[1] Lelièvre-Stoltz-Rousset, *Langevin dynamics with constraints and computation of free energy differences*, 2012

Remark: High Dimensional/Low Temperature Applications

Motivated by pbs where:

$$\text{Relative Entropy}(\text{target} \mid \text{educated proposal}) \to +\infty.$$

(that is Importance Sampling fails, see also Chatterjee, Diaconis AAP 2018) .

Remark: High Dimensional/Low Temperature Applications

Motivated by pbs where:

$$\text{Relative Entropy(target} \mid \text{educated proposal)} \to +\infty.$$

(that is Importance Sampling fails, see also Chatterjee, Diaconis AAP 2018) .

- Sampling w.r.t. low temperature Gibbs distribution. Tempering: $\pi_s \propto \mathrm{e}^{-sU(x)}\pi(dx)$.

Remark: High Dimensional/Low Temperature Applications

Motivated by pbs where:

$$\text{Relative Entropy}(\text{target} \mid \text{educated proposal}) \to +\infty.$$

(that is Importance Sampling fails, see also Chatterjee, Diaconis AAP 2018) .

- Sampling w.r.t. low temperature Gibbs distribution. Tempering: $\pi_s \propto \mathrm{e}^{-sU(x)} \pi(dx)$.
- High dimensional Bayesian statistics: $\pi =$ prior distribution on model(s). $-V(s,x) =$ (smoothed) log-likelihood from $s \times n_{\mathrm{obs}}$ datas.

Remark: High Dimensional/Low Temperature Applications

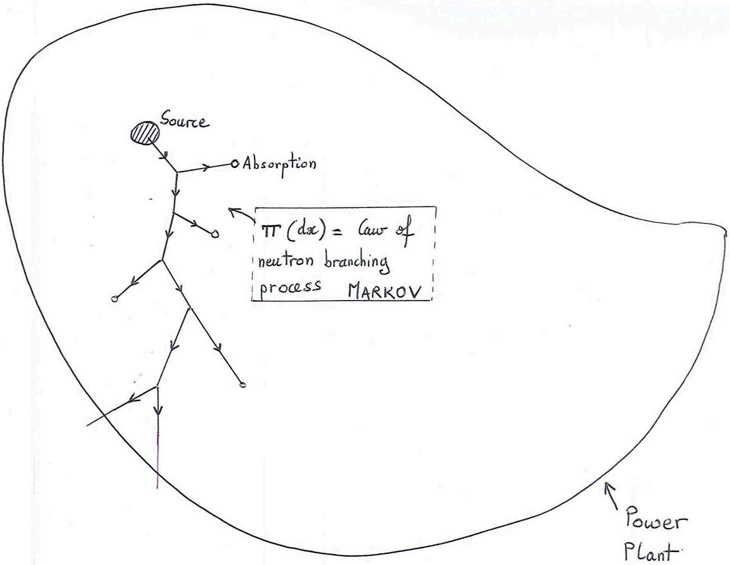
Motivated by pbs where:

$$\text{Relative Entropy(target} \mid \text{educated proposal)} \to +\infty.$$

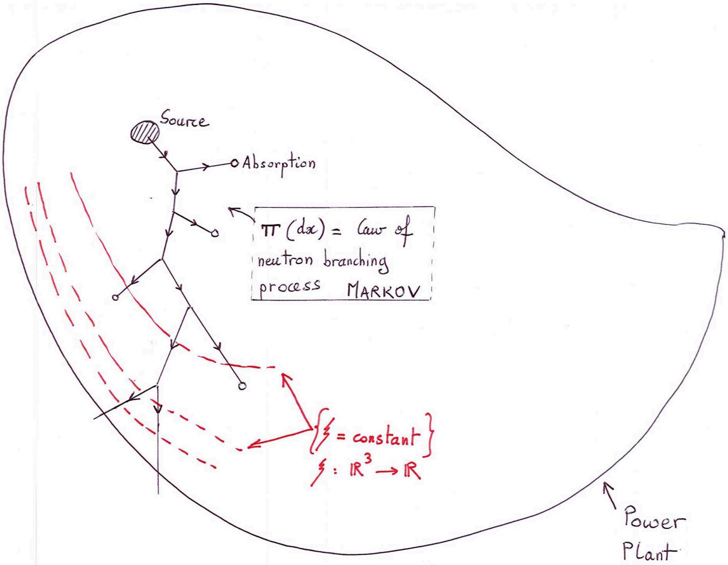
(that is Importance Sampling fails, see also Chatterjee, Diaconis AAP 2018) .

- Sampling w.r.t. low temperature Gibbs distribution. Tempering: $\pi_s \propto \mathrm{e}^{-sU(x)}\pi(dx)$.
- High dimensional Bayesian statistics: $\pi =$ prior distribution on model(s). $-V(s,x) =$ (smoothed) log-likelihood from $s \times n_{\mathrm{obs}}$ datas.
- Rare event.

Branching Neutron Transport Application



Branching Neutron Transport Application



Branching Neutron Transport Application

Rare Event formalism:

Branching Neutron Transport Application

Rare Event formalism:

- Exemple of choice of score function, e_0 given critical energy:

$$\mathrm{score}(X) := \max_{t \geqslant 0} \zeta(X_t)$$

$$= \max_{t \geqslant 0} \inf \Big\{ s \in \mathbb{R} \mid \text{total energy of neutrons at time } t \text{ in } \{\xi \geqslant s\} \leqslant e_0 \Big\}$$

Branching Neutron Transport Application

Rare Event formalism:

- Exemple of choice of score function, e_0 given critical energy:

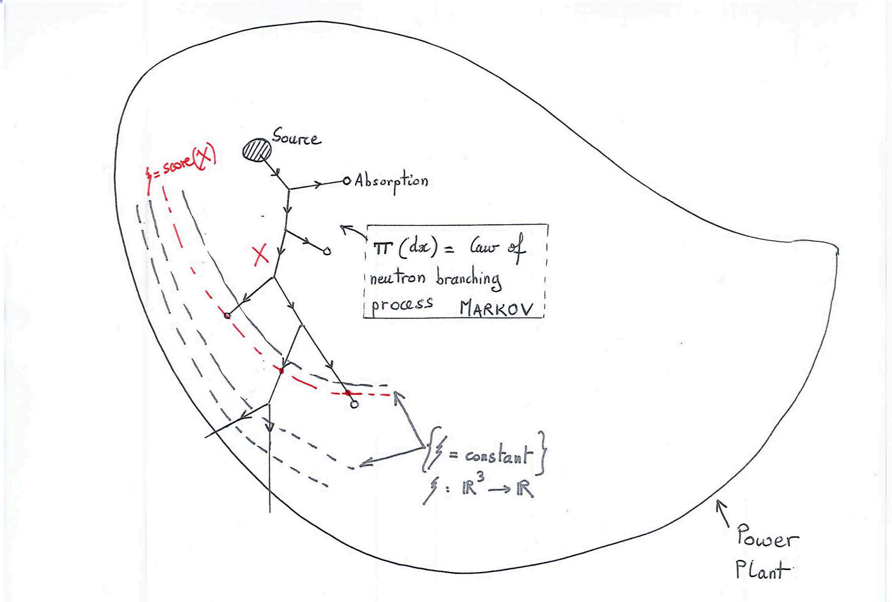
$$\mathrm{score}(X) := \max_{t \geqslant 0} \zeta(X_t)$$

$$= \max_{t \geqslant 0} \inf \Big\{ s \in \mathbb{R} \mid \text{total energy of neutrons at time } t \text{ in } \{\xi \geqslant s\} \leqslant e_0 \Big\}$$

- Corresponding choice of rare event flow:

$$z_s = p_s := \pi(\{\mathrm{score} \geqslant s\}), \qquad \eta_s := \pi(\quad \mid \mathrm{score} \geqslant s)$$

Branching Neutron Transport Application



Branching Neutron Transport Application

Branching Neutron Transport Application

- Define the stopping time for the natural filtration of the branching neutron transport process X

$$\tau_s(X) := \inf\{t \geqslant 0 \mid \zeta(X_t) = s\}$$

Note that

$$\{\mathrm{score}(X) < s\} = \{\tau_s(X) = +\infty\}\,.$$

Branching Neutron Transport Application

- Define the stopping time for the natural filtration of the branching neutron transport process X

$$\tau_s(X) := \inf\{t \geqslant 0 \mid \zeta(X_t) = s\}$$

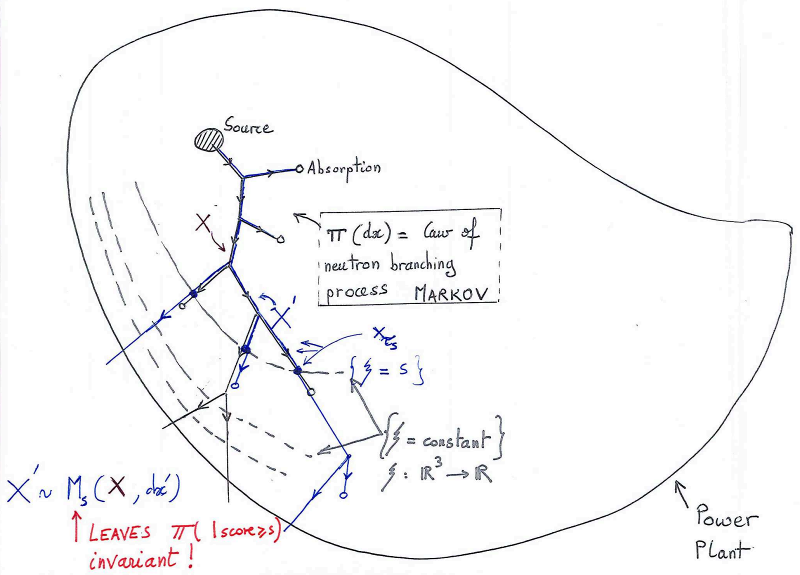
Note that

$$\{\mathrm{score}(X) < s\} = \{\tau_s(X) = +\infty\}\,.$$

- Using the strong Markov property, one can simulate new neutrons starting from $X_{\tau_s(X)}$. This yields a Markov kernel $M_s(x, dx')$ leaving π conditioned by $\{\mathrm{score} \geqslant s\}$ invariant.

$$\int M_s(x, dx')\eta_s(dx) = \eta_s(dx').$$

Branching Neutron Transport Application



Sequential Monte-Carlo a.k.a. Importance Splitting

Define: $0 = s_{(0)} < \ldots < s_{(i_{\max})} = 1$ a given, finite ladder of scores.

$X^n_{s_{(i)}}$ state of replica n at iteration i.

General Form of the Algorithm with Weighted Replicas:

Sequential Monte-Carlo a.k.a. Importance Splitting

Define: $0 = s_{(0)} < \ldots < s_{(i_{\max})} = 1$ a given, finite ladder of scores.

$X^n_{s_{(i)}}$ state of replica n at iteration i.

General Form of the Algorithm with Weighted Replicas:

- (0) Simulate N independent replicas according to η_0.

Iterate on $i = 1 \ldots i_{\max}$:

Sequential Monte-Carlo a.k.a. Importance Splitting

Define: $0 = s_{(0)} < \ldots < s_{(i_{\max})} = 1$ a given, finite ladder of scores.

$X^n_{s_{(i)}}$ state of replica n at iteration i.

General Form of the Algorithm with Weighted Replicas:

- (0) Simulate N independent replicas according to η_0.

Iterate on $i = 1 \ldots i_{\max}$:

- (i) **Weights:** update the 'importance weight' of each replica $n \in (1, N)$ by $\mathrm{e}^{-V_{s_{(i)}}(X^n_{s_{(i-1)}})+V_{s_{(i-1)}}(X^n_{s_{(i-1)}})}$ (target: $\mathrm{e}^{-V_{s_{(i)}}}\pi$).

Sequential Monte-Carlo a.k.a. Importance Splitting

Define: $0 = s_{(0)} < \ldots < s_{(i_{\max})} = 1$ a given, finite ladder of scores.

$X^n_{s_{(i)}}$ state of replica n at iteration i.

General Form of the Algorithm with Weighted Replicas:

- (0) Simulate N independent replicas according to η_0.

Iterate on $i = 1 \ldots i_{\max}$:

- (i) **Weights:** update the 'importance weight' of each replica $n \in (1, N)$ by $\mathrm{e}^{-V_{s_{(i)}}(X^n_{s_{(i-1)}}) + V_{s_{(i-1)}}(X^n_{s_{(i-1)}})}$ (target: $\mathrm{e}^{-V_{s_{(i)}}}\pi$).
- (ii) **Selection (optional)** kill and/or split replicas and update weights. E.g.: triggered if weights are too degenerate.

Sequential Monte-Carlo a.k.a. Importance Splitting

Define: $0 = s_{(0)} < \ldots < s_{(i_{\max})} = 1$ a given, finite ladder of scores.

$X^n_{s_{(i)}}$ state of replica n at iteration i.

General Form of the Algorithm with Weighted Replicas:

- (0) Simulate N independent replicas according to η_0.

Iterate on $i = 1 \ldots i_{\max}$:

- (i) **Weights:** update the 'importance weight' of each replica $n \in (1, N)$ by $\mathrm{e}^{-V_{s_{(i)}}(X^n_{s_{(i-1)}}) + V_{s_{(i-1)}}(X^n_{s_{(i-1)}})}$ (target: $\mathrm{e}^{-V_{s_{(i)}}} \pi$).
- (ii) **Selection (optional)** kill and/or split replicas and update weights. E.g.: triggered if weights are too degenerate.
- (iii) **Mutation (optional)**: modify ('mutate') (all or some or none) replicas with Markov Chain Monte Carlo transition $M_{s_{(i)}}(x, dx')$ that leaves invariant the target $\eta_{s_{(i)}}(dx)$.

Sequential Monte-Carlo a.k.a. Importance Splitting

Estimators:

Sequential Monte-Carlo a.k.a. Importance Splitting

Estimators:

- Target measures $\eta_s = \frac{1}{z_s} \mathrm{e}^{-V(x,s)} \pi(dx)$ are estimated by weighted empirical measures with normalization

$$\eta^N_{s_{(i)}} := \sum_{n=1}^{N} \mathrm{Weight}^n_{s_{(i)}} \delta_{X^n_{s_{(i)}}} \Big/ \sum_{n=1}^{N} \mathrm{Weight}^n_{s_{(i)}} .$$

Sequential Monte-Carlo a.k.a. Importance Splitting

Estimators:

- Target measures $\eta_s = \frac{1}{z_s} \mathrm{e}^{-V(x,s)} \pi(dx)$ are estimated by weighted empirical measures with normalization

$$\eta^N_{s_{(i)}} := \sum_{n=1}^{N} \mathrm{Weight}^n_{s_{(i)}} \delta_{X^n_{s_{(i)}}} \Big/ \sum_{n=1}^{N} \mathrm{Weight}^n_{s_{(i)}} .$$

- Normalizations are estimated by the average weights over replicas

$$z^N_{s_{(i)}} := \frac{1}{N} \sum_{n=1}^{N} \mathrm{Weight}^n_{s_{(i)}}$$

Selection schemes

Definition

A selection or re-sampling scheme draw branching numbers $B_n \in \mathbb{N}$ splitting or killing each replica $n = 1 \dots N$. New weights are defined accordingly:

$$\mathbb{E}[\mathrm{Weight}^n_{new} B^n \mid \mathrm{Weight}] = \mathrm{Weight}^n$$

so that the Effective Sample Size (e.g. Renyi entropy between weighted and unweighted empirical distrib.) increases.

Selection schemes

Definition

A selection or re-sampling scheme draw branching numbers $B_n \in \mathbb{N}$ splitting or killing each replica $n = 1 \dots N$. New weights are defined accordingly:

$$\mathbb{E}[\mathrm{Weight}^n_{new} B^n \mid \mathrm{Weight}] = \mathrm{Weight}^n$$

so that the Effective Sample Size (e.g. Renyi entropy between weighted and unweighted empirical distrib.) increases.

- $B^n \geqslant 1$: *selection of* **splitting type**.

Selection schemes

Definition

A selection or re-sampling scheme draw branching numbers $B_n \in \mathbb{N}$ *splitting or killing each replica* $n = 1 \dots N$. *New weights are defined accordingly:*

$$\mathbb{E}[\mathrm{Weight}^n_{new} B^n \mid \mathrm{Weight}] = \mathrm{Weight}^n$$

so that the Effective Sample Size (e.g. Renyi entropy between weighted and unweighted empirical distrib.) increases.

- $B^n \geqslant 1$: *selection of* **splitting type**.
- $B^n \leqslant 1$: *selection of* **killing type**.

Selection schemes

Definition

A selection or re-sampling scheme draw branching numbers $B_n \in \mathbb{N}$ splitting or killing each replica $n = 1 \dots N$. New weights are defined accordingly:

$$\mathbb{E}[\mathrm{Weight}^n_{new} B^n \mid \mathrm{Weight}] = \mathrm{Weight}^n$$

so that the Effective Sample Size (e.g. Renyi entropy between weighted and unweighted empirical distrib.) increases.

- $B^n \geqslant 1$ *: selection of* **splitting type**.
- $B^n \leqslant 1$ *: selection of* **killing type**.
- $B^n \geqslant 1$ *and* $\mathbb{E}(B_n)$ *is independent on* n*:* **neutral bearing**.

Basic Refs

Papers:

Basic Refs

Papers:

- Del Moral Doucet Jasra *Sequential Monte Carlo samplers* 2006.

Basic Refs

Papers:

- Del Moral Doucet Jasra *Sequential Monte Carlo samplers* 2006.
- A Beskos, A Jasra, N Kantas, A Thiery *On the convergence of adaptive sequential Monte Carlo methods* 2016

Basic Refs

Papers:

- Del Moral Doucet Jasra *Sequential Monte Carlo samplers* 2006.
- A Beskos, A Jasra, N Kantas, A Thiery *On the convergence of adaptive sequential Monte Carlo methods* 2016
- F Cérou, P Del Moral, T Furon, A Guyader *Sequential Monte Carlo for rare event estimation* 2012

Basic Refs

Papers:

- Del Moral Doucet Jasra *Sequential Monte Carlo samplers* 2006.
- A Beskos, A Jasra, N Kantas, A Thiery *On the convergence of adaptive sequential Monte Carlo methods* 2016
- F Cérou, P Del Moral, T Furon, A Guyader *Sequential Monte Carlo for rare event estimation* 2012
- F Cérou, A Guyader, *Adaptive Multilevel Splitting for rare event analysis*, 2007.

Basic Refs

Papers:

- Del Moral Doucet Jasra *Sequential Monte Carlo samplers* 2006.
- A Beskos, A Jasra, N Kantas, A Thiery *On the convergence of adaptive sequential Monte Carlo methods* 2016
- F Cérou, P Del Moral, T Furon, A Guyader *Sequential Monte Carlo for rare event estimation* 2012
- F Cérou, A Guyader, *Adaptive Multilevel Splitting for rare event analysis*, 2007.
- In Phys.: 'Jarzynski equality'

Basic Refs

Papers:

- Del Moral Doucet Jasra *Sequential Monte Carlo samplers* 2006.
- A Beskos, A Jasra, N Kantas, A Thiery *On the convergence of adaptive sequential Monte Carlo methods* 2016
- F Cérou, P Del Moral, T Furon, A Guyader *Sequential Monte Carlo for rare event estimation* 2012
- F Cérou, A Guyader, *Adaptive Multilevel Splitting for rare event analysis*, 2007.
- In Phys.: 'Jarzynski equality'
- Freddy Bouchet and al..

Books

Basic Refs

Papers:

- Del Moral Doucet Jasra *Sequential Monte Carlo samplers* 2006.
- A Beskos, A Jasra, N Kantas, A Thiery *On the convergence of adaptive sequential Monte Carlo methods* 2016
- F Cérou, P Del Moral, T Furon, A Guyader *Sequential Monte Carlo for rare event estimation* 2012
- F Cérou, A Guyader, *Adaptive Multilevel Splitting for rare event analysis*, 2007.
- In Phys.: 'Jarzynski equality'
- Freddy Bouchet and al..

Books

- Liu *Monte Carlo Strategies*

Basic Refs

Papers:

- Del Moral Doucet Jasra *Sequential Monte Carlo samplers* 2006.
- A Beskos, A Jasra, N Kantas, A Thiery *On the convergence of adaptive sequential Monte Carlo methods* 2016
- F Cérou, P Del Moral, T Furon, A Guyader *Sequential Monte Carlo for rare event estimation* 2012
- F Cérou, A Guyader, *Adaptive Multilevel Splitting for rare event analysis*, 2007.
- In Phys.: 'Jarzynski equality'
- Freddy Bouchet and al..

Books

- Liu *Monte Carlo Strategies*
- Chopin *Introduction To Sequential Monte Carlo*

Basic Refs

Papers:

- Del Moral Doucet Jasra *Sequential Monte Carlo samplers* 2006.
- A Beskos, A Jasra, N Kantas, A Thiery *On the convergence of adaptive sequential Monte Carlo methods* 2016
- F Cérou, P Del Moral, T Furon, A Guyader *Sequential Monte Carlo for rare event estimation* 2012
- F Cérou, A Guyader, *Adaptive Multilevel Splitting for rare event analysis*, 2007.
- In Phys.: 'Jarzynski equality'
- Freddy Bouchet and al..

Books

- Liu *Monte Carlo Strategies*
- Chopin *Introduction To Sequential Monte Carlo*
- Doucet, Freitas, Gordon *Sequential Monte Carlo in Practice*

Basic Refs

Papers:

- Del Moral Doucet Jasra *Sequential Monte Carlo samplers* 2006.
- A Beskos, A Jasra, N Kantas, A Thiery *On the convergence of adaptive sequential Monte Carlo methods* 2016
- F Cérou, P Del Moral, T Furon, A Guyader *Sequential Monte Carlo for rare event estimation* 2012
- F Cérou, A Guyader, *Adaptive Multilevel Splitting for rare event analysis*, 2007.
- In Phys.: 'Jarzynski equality'
- Freddy Bouchet and al..

Books

- Liu *Monte Carlo Strategies*
- Chopin *Introduction To Sequential Monte Carlo*
- Doucet, Freitas, Gordon *Sequential Monte Carlo in Practice*
- Del Moral *Feynman-Kac formula*

What is 'adaptivity' ?

What is 'adaptivity' ?

- A 'non-adaptive' SMC/Importance Splitting algorithm consist of: i) preset ladder of scores $0 = s_{(0)} < \ldots < s_{(i_{\max})} = 1$, ii) preset choice of mutations M_s leaving target η_s invariant.

What is 'adaptivity' ?

- A 'non-adaptive' SMC/Importance Splitting algorithm consist of: i) preset ladder of scores $0 = s_{(0)} < \ldots < s_{(i_{\max})} = 1$, ii) preset choice of mutations M_s leaving target η_s invariant.
- Many 'adaptive' variants (e.g. Adaptive Multilevel Splitting, see after) are presented as follows: the choice of the scores is random, adaptive.

What is 'adaptivity' ?

- A 'non-adaptive' SMC/Importance Splitting algorithm consist of: i) preset ladder of scores $0 = s_{(0)} < \ldots < s_{(i_{\max})} = 1$, ii) preset choice of mutations M_s leaving target η_s invariant.
- Many 'adaptive' variants (e.g. Adaptive Multilevel Splitting, see after) are presented as follows: the choice of the scores is random, adaptive.
- In this talk I propose a different 'mindset':

Idea

Scores are always deterministic -> continuous ladder.

Interpret **'Adaptivity'** $=$ **'Mutations Only If Selection'**.

Nothing happens for most scores ('weighting $=$ waiting') because of the triggering of mutations by selection.

Triggered Mutations

Consider the mutation M_s after the selection step in the algo.
Vocabulary:

Triggered Mutations

Consider the mutation M_s after the selection step in the algo.
Vocabulary:

- **Preset Mutations**: M_s is preset, applied to all replicas at each score → non-adaptive, 'Feynman-Kac-Del Moral structure'.

Triggered Mutations

Consider the mutation M_s after the selection step in the algo.
Vocabulary:

- **Preset Mutations**: M_s is preset, applied to all replicas at each score $\to$ non-adaptive, 'Feynman-Kac-Del Moral structure'.
- **Mutations-If-Selection**: A mutation kernel M_s is triggered only when selection step is triggered.

Triggered Mutations

Consider the mutation M_s after the selection step in the algo.
Vocabulary:

- **Preset Mutations**: M_s is preset, applied to all replicas at each score $\to$ non-adaptive, 'Feynman-Kac-Del Moral structure'.
- **Mutations-If-Selection**: A mutation kernel M_s is triggered only when selection step is triggered.
- **Mutations-On-Child**: The mutation kernel M_s applied only to children when selction is such that the sample of children $\simeq$ target.

Triggered Mutations

Example (Mutations-If-Selection)

Triggered Mutations

Example (Mutations-If-Selection)

- *Compute an Effective Sample Size (ESS) of the weights at each score/iteration.*

Triggered Mutations

Example (Mutations-If-Selection)

- *Compute an Effective Sample Size (ESS) of the weights at each score/iteration.*
- *If ESS is above a treshhold: trigger selection.*

Triggered Mutations

Example (Mutations-If-Selection)

- *Compute an Effective Sample Size (ESS) of the weights at each score/iteration.*
- *If ESS is above a treshhold: trigger selection.*
- *If selection happens, mutations on all replicas are triggered.*

Example (Mutations-On-Child)

Triggered Mutations

Example (Mutations-If-Selection)

- *Compute an Effective Sample Size (ESS) of the weights at each score/iteration.*
- *If ESS is above a treshhold: trigger selection.*
- *If selection happens, mutations on all replicas are triggered.*

Example (Mutations-On-Child)

- *Special case of Mutations-If-Selection.*

Triggered Mutations

Example (Mutations-If-Selection)

- *Compute an Effective Sample Size (ESS) of the weights at each score/iteration.*
- *If ESS is above a treshhold: trigger selection.*
- *If selection happens, mutations on all replicas are triggered.*

Example (Mutations-On-Child)

- *Special case of Mutations-If-Selection.*
- *Resampling/selection is such that for splitting:*

$$\mathrm{Law}(Child) = \frac{1}{N} \sum_{n=1}^{N} W^n \delta_{X^n} \simeq \mathrm{target}.$$

Triggered Mutations

Example (Mutations-If-Selection)

- *Compute an Effective Sample Size (ESS) of the weights at each score/iteration.*
- *If ESS is above a treshhold: trigger selection.*
- *If selection happens, mutations on all replicas are triggered.*

Example (Mutations-On-Child)

- *Special case of Mutations-If-Selection.*
- *Resampling/selection is such that for splitting:*

$$\mathrm{Law}(Child) = \frac{1}{N} \sum_{n=1}^{N} W^n \delta_{X^n} \simeq \mathrm{target}.$$

- *Triggered mutations are applied on children ONLY.*

Adaptive/Triggered Mutation variant

Remarks

Adaptive/Triggered Mutation variant

Remarks

- *The goal of triggered Mutations (If-Selection, On-Child) is to save computational power by avoiding mutations (hence evaluation of V or ∇V) if simple weighting is sufficient.*

Adaptive/Triggered Mutation variant

Remarks

- *The goal of triggered Mutations (If-Selection, On-Child) is to save computational power by avoiding mutations (hence evaluation of* V *or* ∇V *) if simple weighting is sufficient.*
- *Consistency of Adaptive mutations: large sample* $N \to +\infty$.

Adaptive/Triggered Mutation variant

Remarks

- *The goal of triggered Mutations (If-Selection, On-Child) is to save computational power by avoiding mutations (hence evaluation of V or ∇V) if simple weighting is sufficient.*
- *Consistency of Adaptive mutations: large sample $N \to +\infty$.*
- *Consistency of Adaptive mutations: large mixing $M_s \to \eta_s$.*

Adaptive/Triggered Mutation variant

Remarks

- *The goal of triggered Mutations (If-Selection, On-Child) is to save computational power by avoiding mutations (hence evaluation of V or ∇V) if simple weighting is sufficient.*
- *Consistency of Adaptive mutations: large sample $N \to +\infty$.*
- *Consistency of Adaptive mutations: large mixing $M_s \to \eta_s$.*
- *Well-known rare event case: Adaptive Multilevel Splitting (AMS) algorithm (see after).*

Adaptive/Triggered Mutation variant

Remarks

- *The goal of triggered Mutations (If-Selection, On-Child) is to save computational power by avoiding mutations (hence evaluation of V or ∇V) if simple weighting is sufficient.*
- *Consistency of Adaptive mutations: large sample $N \to +\infty$.*
- *Consistency of Adaptive mutations: large mixing $M_s \to \eta_s$.*
- *Well-known rare event case: Adaptive Multilevel Splitting (AMS) algorithm (see after).*
- *AMS in the dynamical setting has a hidden non-adaptive Feynman-Kac-Del Moral structure (see below).*

Remark on Adaptive Mutations

Additional Intrinsic Adaptivity on Mutations: The mutation kernel M_s is random and depends on the past replica empirical distribution. E.g.: if M_s is based on accept/reject, proposal is adaptively tuned to target an average acceptance rate $r_0 \in (0, 1)$.

The Feynman-Kac-Del Moral structure

The Feynman-Kac-Del Moral structure

- For non-adaptive = preset mutations, the algorithm can be derived from a Feynman-Kac (FK) formula:

$$\int \varphi(x)\mathrm{e}^{-V_{s_{(i)}}(x)}\pi(dx) =$$

$$\mathbb{E}\left[\varphi(X_{s_{(i)}})\mathrm{e}^{-\sum_{i'=1}^{i} V_{s_{(i')}}(X_{s_{(i'-1)}})-V_{s_{(i'-1)}}(X_{s_{(i'-1)}})}\right]$$

where $X_{s_{(i)}}$, $i \geqslant 0$ is a Markov chain with $X_0 \sim \eta_0$ and probability transition $M_{s_{(i)}}$.

The Feynman-Kac-Del Moral structure

- For non-adaptive = preset mutations, the algorithm can be derived from a Feynman-Kac (FK) formula:

$$\int \varphi(x) e^{-V_{s_{(i)}}(x)} \pi(dx) = \mathbb{E}\left[\varphi(X_{s_{(i)}}) e^{-\sum_{i'=1}^{i} V_{s_{(i')}}(X_{s_{(i'-1)}}) - V_{s_{(i'-1)}}(X_{s_{(i'-1)}})} \right]$$

  where $X_{s_{(i)}}$, $i \geqslant 0$ is a Markov chain with $X_0 \sim \eta_0$ and probability transition $M_{s_{(i)}}$.
- The algoritm is then: simulating independently N chains with FK weights. Additional re-sampling/selection to prevent weight degeneracy.

The Feynman-Kac-Del Moral structure

- For non-adaptive = preset mutations, the algorithm can be derived from a Feynman-Kac (FK) formula:

$$\int \varphi(x) \mathrm{e}^{-V_{s_{(i)}}(x)} \pi(dx) =$$

$$\mathbb{E}\left[\varphi(X_{s_{(i)}}) \mathrm{e}^{-\sum_{i'=1}^{i} V_{s_{(i')}}(X_{s_{(i'-1)}}) - V_{s_{(i'-1)}}(X_{s_{(i'-1)}})} \right]$$

  where $X_{s_{(i)}}$, $i \geqslant 0$ is a Markov chain with $X_0 \sim \eta_0$ and probability transition $M_{s_{(i)}}$.

- The algoritm is then: simulating independently N chains with FK weights. Additional re-sampling/selection to prevent weight degeneracy.
- Nota Bene: in Del Moral, re-sampling/selection is put in a (very slightly restrictive) 'mean-field' form.

Jarzynski equality

Remark

The Feynman-Kac formula before is known in physics as 'Jarzynski equality'. In that case:

Jarzynski equality

Remark

The Feynman-Kac formula before is known in physics as 'Jarzynski equality'. In that case:

- *s is reaction coordinate or a thermodynamic parameter.*

Jarzynski equality

Remark

The Feynman-Kac formula before is known in physics as 'Jarzynski equality'. In that case:

- *s is reaction coordinate or a thermodynamic parameter.*
- *Target is a canonical Gibbs distribution (mechanical system thermostatted).*

Jarzynski equality

Remark

The Feynman-Kac formula before is known in physics as 'Jarzynski equality'. In that case:

- *s is reaction coordinate or a thermodynamic parameter.*
- *Target is a canonical Gibbs distribution (mechanical system thermostatted).*
- *Mutation is Newton dynamics with parameter s + random perturbation at given temperature (Langevin).*

Jarzynski equality

Remark

The Feynman-Kac formula before is known in physics as 'Jarzynski equality'. In that case:

- *s is reaction coordinate or a thermodynamic parameter.*
- *Target is a canonical Gibbs distribution (mechanical system thermostatted).*
- *Mutation is Newton dynamics with parameter s + random perturbation at given temperature (Langevin).*
- *Weight $= e^{-\mathrm{Work}/(k_b T)}$!!*

Jarzynski equality

Remark

The Feynman-Kac formula before is known in physics as 'Jarzynski equality'. In that case:

- *s is reaction coordinate or a thermodynamic parameter.*
- *Target is a canonical Gibbs distribution (mechanical system thermostatted).*
- *Mutation is Newton dynamics with parameter s + random perturbation at given temperature (Langevin).*
- *Weight $= e^{-\mathrm{Work}/(k_b T)}$!!*
- *Exists experimentally !!*

The Feynman-Kac-Del Moral structure

Proposition (Unbiasedness)

Un-normalized estimators are unbiased for algorithms following the Feynman-Kac-Del Moral structure.

Proof.

First remark that $\int \varphi \, \mathrm{e}^{-V_{s(i)}} \, d\pi = \mathbb{E}[\varphi(X_{s_{(i)}})\mathrm{e}^{-V_{s(i)}(X_{s_{(i-1)}})+V_{s(i-1)}(X_{s_{(i-1)}})} \times \ldots \times \mathrm{e}^{-V_{s(1)}(X_{s(0)})+V_{s(0)}(X_{s(0)})}] =: \mathbb{E}[Q^{0\to i}(\varphi)(X_0)]$ where $i \mapsto X_{(i)}$ is the MCMC chain used in the mutation step. Then check that for $i \leqslant i_0$

$$i \mapsto z^N_{s(i)} \int Q^{i\to i^0}(\varphi) \, d\eta^N_{s(i)}$$ is a martingale.

$\square$

High dimension requires continuous

High dimension requires continuous

- High Dimension $d \gg 1$: weights that are $\times$ by $\mathrm{e}^{-V_{s_{(i+1)}}(X_{s_{(i)}})+V_{s_{(i)}}(X_{s_{(i)}})}$ at each iteration have exponential variance with d (typically).

Example

In $\mathbb{R}^d$, if coordinates of X are i.i.d. and V has a sum form over coordinates and is smooth w.r.t. s, by CLT, non-degeneracy of weights requires:

$$s^{(i+1)} - s^{(i)} \sim \frac{1}{\sqrt{d}} \xrightarrow{d \to +\infty} 0.$$

High dimension requires continuous

- High Dimension $d \gg 1$: weights that are $\times$ by $\mathrm{e}^{-V_{s_{(i+1)}}(X_{s_{(i)}})+V_{s_{(i)}}(X_{s_{(i)}})}$ at each iteration have exponential variance with d (typically).

Example

In $\mathbb{R}^d$, if coordinates of X are i.i.d. and V has a sum form over coordinates and is smooth w.r.t. s, by CLT, non-degeneracy of weights requires:

$$s^{(i+1)} - s^{(i)} \sim \frac{1}{\sqrt{d}} \xrightarrow{d \to +\infty} 0.$$

- Tempting to not mutate at each $s^{(i)}$.

High dimension requires continuous

- High Dimension $d \gg 1$: weights that are $\times$ by $\mathrm{e}^{-V_{s_{(i+1)}}(X_{s_{(i)}})+V_{s_{(i)}}(X_{s_{(i)}})}$ at each iteration have exponential variance with d (typically).

Example

In $\mathbb{R}^d$, if coordinates of X are i.i.d. and V has a sum form over coordinates and is smooth w.r.t. s, by CLT, non-degeneracy of weights requires:

$$s^{(i+1)} - s^{(i)} \sim \frac{1}{\sqrt{d}} \xrightarrow{d\to+\infty} 0.$$

- Tempting to not mutate at each $s^{(i)}$.
- Idea: switch to a continuum of scores:

$$s \in \left\{ s^{(0)}, \dots, s^{(I)} \right\} \quad \text{becomes} \quad s \in [0,1].$$

Indexing the algorithm by selection events

'Same' algorithm, new representation:

Indexing the algorithm by selection events

'Same' algorithm, new representation:

- **Preset Mutations**: Each replicas evolve independently according to a Markov process with generator L_s invariant with respect to target $\eta_s \propto \mathrm{e}^{-V_s}\pi$.

Example

Piecewise constant Markov jump process

$$L_s(\varphi)(x) = \lambda_s(M_s(\varphi)(x) - \varphi(x)), \quad \eta_s M_s = \eta_s$$

can be simulated: i) mutations occur at random score (higher than s_0 with proba $\mathrm{e}^{-\int_0^{s_0} \lambda_s ds}$), ii) mutations with M_s.

Indexing the algorithm by selection events

'Same' algorithm, new representation:

- **Preset Mutations**: Each replicas evolve independently according to a Markov process with generator L_s invariant with respect to target $\eta_s \propto \mathrm{e}^{-V_s}\pi$.

Example

Piecewise constant Markov jump process

$$L_s(\varphi)(x) = \lambda_s(M_s(\varphi)(x) - \varphi(x)), \quad \eta_s M_s = \eta_s$$

can be simulated: i) mutations occur at random score (higher than s_0 with proba $\mathrm{e}^{-\int_0^{s_0} \lambda_s ds}$), ii) mutations with M_s.

- Other examples: discretization of a Stochastic Differential Equation, or Piecewise Deterministic Markov Process.

Re-Indexing the algorithm by splitting events

Initialize replicas and set $S_{(0)} = 0$. Mutate all particles with L_s on $s \in [0, 1]$. **Iterate on j**:

Re-Indexing the algorithm by splitting events

Initialize replicas and set $S_{(0)} = 0$. Mutate all particles with L_s on $s \in [0,1]$. **Iterate on j:**

- (j) **Weights:** compute the 'importance' weight for $s \in [0,1]$ of replicas so that it targets η_s for each s, e.g.: $\mathrm{e}^{-\int_0^s \partial_{s'} V_{s'}(X_{s'}) ds'}$.

Re-Indexing the algorithm by splitting events

Initialize replicas and set $S_{(0)} = 0$. Mutate all particles with L_s on $s \in [0,1]$. **Iterate on j:**

- (j) **Weights:** compute the 'importance' weight for $s \in [0,1]$ of replicas so that it targets η_s for each s, e.g.: $\mathrm{e}^{-\int_0^s \partial_{s'} V_{s'}(X_{s'}) ds'}$.
- (j) **Selection** Compute the next random score

$$S_{(j)} := \inf \left\{ s \geqslant S_{(j-1)} | \mathrm{Criteria}_s^N == 1 \right\}$$

e.g.: $\mathrm{Criteria}_s$ = weight degeneracy (Effective Sample Size) at s.

Then perform selection/re-sampling according to weights.

Re-Indexing the algorithm by splitting events

Initialize replicas and set $S_{(0)} = 0$. Mutate all particles with L_s on $s \in [0,1]$. **Iterate on j**:

- (j) **Weights:** compute the 'importance' weight for $s \in [0,1]$ of replicas so that it targets η_s for each s, e.g.: $\mathrm{e}^{-\int_0^s \partial_{s'} V_{s'}(X_{s'})ds'}$.
- (j) **Selection** Compute the next random score

$$S_{(j)} := \inf \left\{ s \geqslant S_{(j-1)} | \text{Criteria}_s^N == 1 \right\}$$

  e.g.: Criteria_s = weight degeneracy (Effective Sample Size) at s.

  Then perform selection/re-sampling according to weights.
- (j) **Triggered Mutations**: additional Mutations-If-Selection with $M_{S_{(j)}}$ (option: On-Child).

Re-Indexing the algorithm by splitting events

Initialize replicas and set $S_{(0)} = 0$. Mutate all particles with L_s on $s \in [0,1]$. **Iterate on j:**

- (j) **Weights:** compute the 'importance' weight for $s \in [0,1]$ of replicas so that it targets η_s for each s, e.g.: $\mathrm{e}^{-\int_0^s \partial_{s'} V_{s'}(X_{s'}) ds'}$.
- (j) **Selection** Compute the next random score

$$S_{(j)} := \inf \left\{ s \geqslant S_{(j-1)} | \text{Criteria}_s^N == 1 \right\}$$

  e.g.: Criteria_s = weight degeneracy (Effective Sample Size) at s.

  Then perform selection/re-sampling according to weights.
- (j) **Triggered Mutations**: additional Mutations-If-Selection with $M_{S_{(j)}}$ (option: On-Child).
- (j) **Preset Mutations**: mutate with L_s on $s \in [S^{(j)}, 1]$ new ($\Leftrightarrow$ all !) replicas.

Re-Indexing the algorithm by splitting events

Initialize replicas and set $S_{(0)} = 0$. Mutate all particles with L_s on $s \in [0,1]$. **Iterate on j:**

- (j) **Weights:** compute the 'importance' weight for $s \in [0,1]$ of replicas so that it targets η_s for each s, e.g.: $\mathrm{e}^{-\int_0^s \partial_{s'} V_{s'}(X_{s'}) ds'}$.
- (j) **Selection** Compute the next random score

$$S_{(j)} := \inf \left\{ s \geqslant S_{(j-1)} | \mathrm{Criteria}_s^N == 1 \right\}$$

  e.g.: $\mathrm{Criteria}_s$ = weight degeneracy (Effective Sample Size) at s.

  Then perform selection/re-sampling according to weights.
- (j) **Triggered Mutations**: additional Mutations-If-Selection with $M_{S_{(j)}}$ (option: On-Child).
- (j) **Preset Mutations**: mutate with L_s on $s \in [S^{(j)}, 1]$ new ($\Leftrightarrow$ all !) replicas.
- (Exit) Stop if $S^{(j)} = 1$ else $j \to j+1$.

Re-Indexing the algorithm by splitting events

Remarks

Re-Indexing the algorithm by splitting events

Remarks

- *Preset mutations are simulated by ANTICIPATION (can be adjusted to decrease cost).*

Re-Indexing the algorithm by splitting events

Remarks

- *Preset mutations are simulated by ANTICIPATION (can be adjusted to decrease cost).*
- *Unbiasedness/Feynman-Kac/Del Moral structure*[a] *holds if no Triggered-Mutation .*

Re-Indexing the algorithm by splitting events

Remarks

- *Preset mutations are simulated by ANTICIPATION (can be adjusted to decrease cost).*
- *Unbiasedness/Feynman-Kac/Del Moral structure[a] holds if no Triggered-Mutation .*
- *AMS in 'static setting' is an example with ONLY Triggered Mutations-On-Child (see after).*

Re-Indexing the algorithm by splitting events

Remarks

- *Preset mutations are simulated by ANTICIPATION (can be adjusted to decrease cost).*
- *Unbiasedness/Feynman-Kac/Del Moral structure*[a] *holds if no Triggered-Mutation .*
- *AMS in 'static setting' is an example with ONLY Triggered Mutations-On-Child (see after).*
- *AMS in 'dynamic setting' is an example with PSEUDO-triggered Mutation-On-Child: they are in fact preset mutations given by the model itself !, (see after).*

[a] See also Brehier Gazeau Goudenege Lelievre Rousset GAMS 2016

Static[2] AMS algorithm

Let $k < N$ given. Assume rare event setting with:

[2] F Cérou, P Del Moral, T Furon, A Guyader *Sequential Monte Carlo for rare event estimation* 2012

Static[2] AMS algorithm

Let $k < N$ given. Assume rare event setting with:

- $\pi :=$ anything simulable.

[2] F Cérou, P Del Moral, T Furon, A Guyader *Sequential Monte Carlo for rare event estimation* 2012

Static[2] AMS algorithm

Let $k < N$ given. Assume rare event setting with:

- $\pi :=$ anything simulable.
- $e^{-V_s} = \mathbf{1}_{\text{score}>s}$.

[2] F Cérou, P Del Moral, T Furon, A Guyader *Sequential Monte Carlo for rare event estimation* 2012

Static[2] AMS algorithm

Let $k < N$ given. Assume rare event setting with:

- $\pi :=$ anything simulable.
- $e^{-V_s} = \mathbf{1}_{\mathsf{score}>s}$.
- $L_s = 0$, only triggered mutations.

[2] F Cérou, P Del Moral, T Furon, A Guyader *Sequential Monte Carlo for rare event estimation* 2012

Static[2] AMS algorithm

Let $k < N$ given. Assume rare event setting with:

- $\pi :=$ anything simulable.
- $\mathrm{e}^{-V_s} = \mathbf{1}_{\text{score}>s}$.
- $L_s = 0$, only triggered mutations.
- Selection = killing + neutral bearing. Triggered by k replicas with lowest score which are killed and then neutrally borne.

[2] F Cérou, P Del Moral, T Furon, A Guyader *Sequential Monte Carlo for rare event estimation* 2012

Static[2] AMS algorithm

Let $k < N$ given. Assume rare event setting with:

- $\pi :=$ anything simulable.
- $\mathrm{e}^{-V_s} = \mathbf{1}_{\mathsf{score} > s}$.
- $L_s = 0$, only triggered mutations.
- Selection = killing + neutral bearing. Triggered by k replicas with lowest score which are killed and then neutrally borne.
- Mutation-If-Selection with Mutation-On-Child. M_s is a MCMC kernel reversible w.r.t. π with rejection if proposal has score $\leqslant s$.

[2] F Cérou, P Del Moral, T Furon, A Guyader *Sequential Monte Carlo for rare event estimation* 2012

Dynamical[3] AMS algorithm

[3] F Cérou, A Guyader, *Adaptive multilevel splitting for rare event analysis*

Dynamical[3] AMS algorithm

- π = Law of a Markov chain / process.

[3] F Cérou, A Guyader, *Adaptive multilevel splitting for rare event analysis*

Dynamical[3] AMS algorithm

- π = Law of a Markov chain / process.
- $\mathrm{e}^{-V_s} = \mathbf{1}_{\text{score}>s}$, score = max($\zeta$(path)).

[3] F Cérou, A Guyader, *Adaptive multilevel splitting for rare event analysis*

Dynamical[3] AMS algorithm

- π = Law of a Markov chain / process.
- $\mathrm{e}^{-V_s} = \mathbf{1}_{\text{score}>s}$, score = max($\zeta$(path)).
- Preset mutations = L_s = Markov generator of π **starting from first hitting time** τ_s **of** $\{\xi > s\}$ to hitting time of $\tau_{s+\delta s}$.

[3] F Cérou, A Guyader, *Adaptive multilevel splitting for rare event analysis*

Dynamical[3] AMS algorithm

- π = Law of a Markov chain / process.
- $\mathrm{e}^{-V_s} = \mathbf{1}_{\text{score}>s}$, score = $\max(\zeta(\text{path}))$.
- Preset mutations = L_s = Markov generator of π **starting from first hitting time** τ_s **of** $\{\xi > s\}$ to hitting time of $\tau_{s+\delta s}$.
- Selection = killing followed by neutral bearing.

[3] F Cérou, A Guyader, *Adaptive multilevel splitting for rare event analysis*

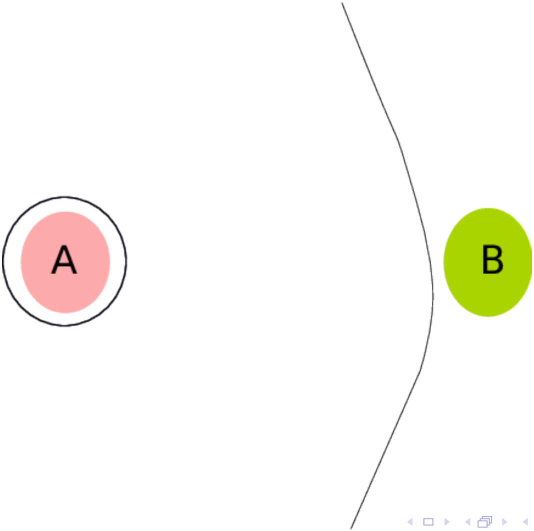
Dynamical[3] AMS algorithm

- π = Law of a Markov chain / process.
- $\mathrm{e}^{-V_s} = \mathbf{1}_{\text{score}>s}$, score = max($\zeta$(path)).
- Preset mutations = L_s = Markov generator of π **starting from first hitting time** τ_s **of** $\{\xi > s\}$ to hitting time of $\tau_{s+\delta s}$.
- Selection = killing followed by neutral bearing.
- The algorithm is just a Fleming-Viot process indexed by s.

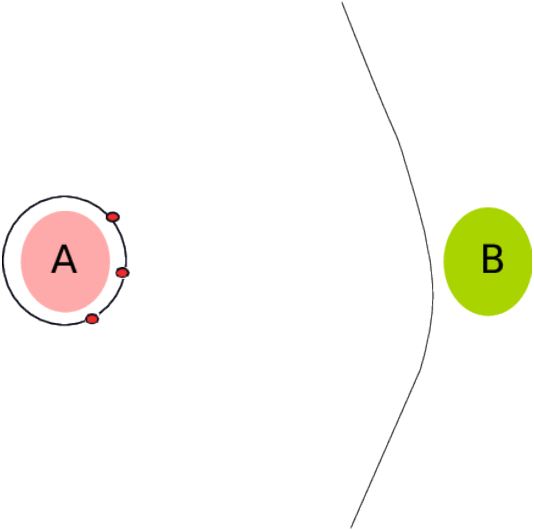
[3] F Cérou, A Guyader, *Adaptive multilevel splitting for rare event analysis*

Adaptive Multilevel Splitting

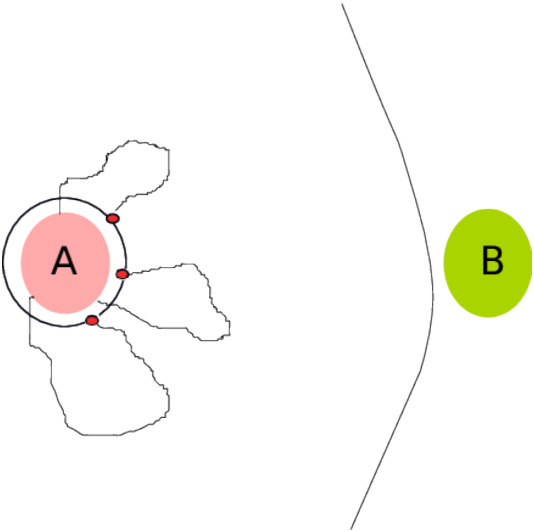
– Black line: $\{\xi = constant\}$.



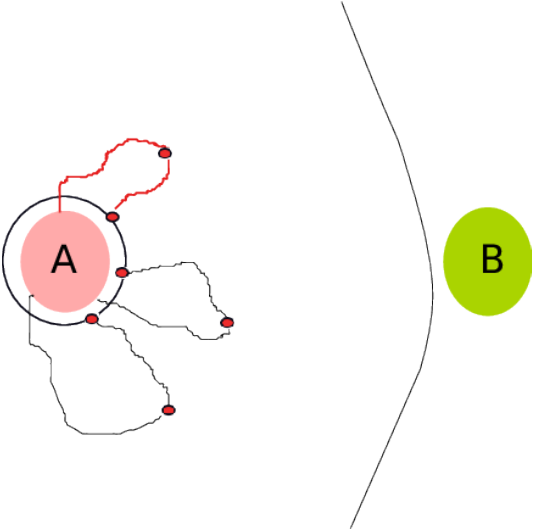
Adaptive Multilevel Splitting



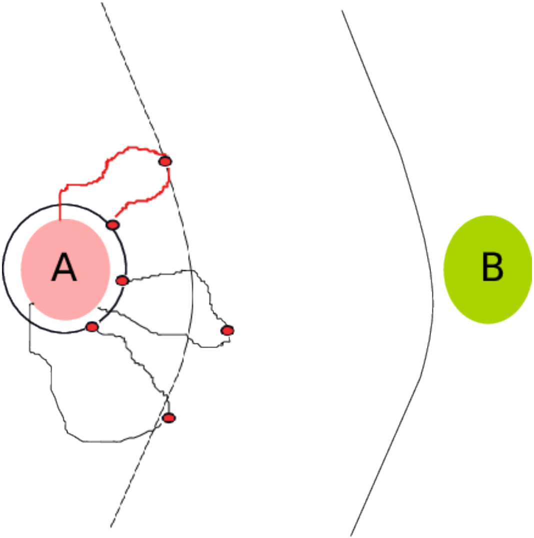
Adaptive Multilevel Splitting



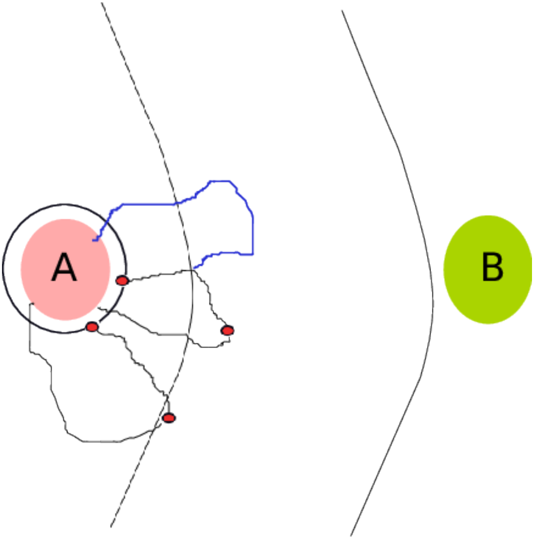
Adaptive Multilevel Splitting



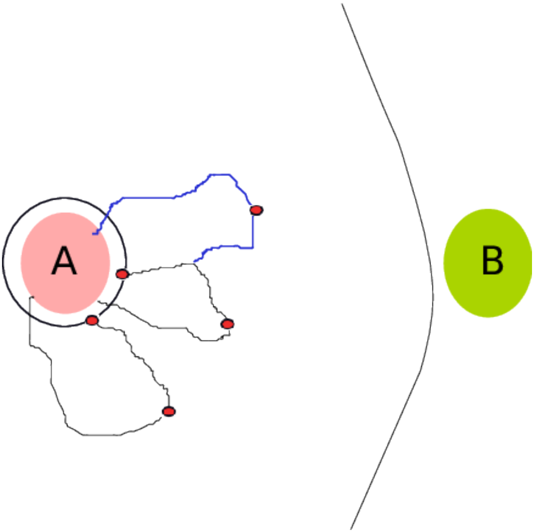
Adaptive Multilevel Splitting



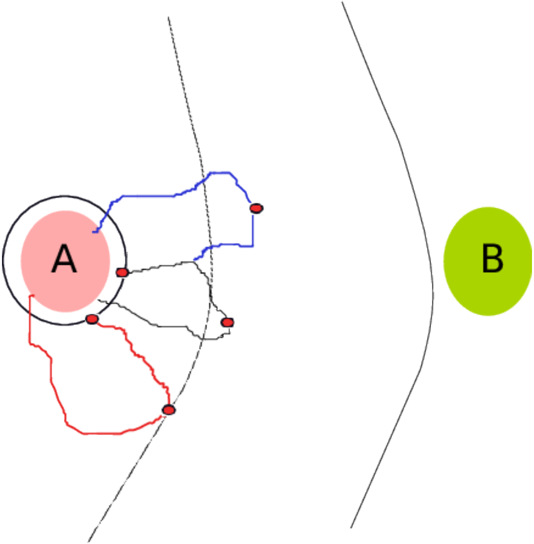
Adaptive Multilevel Splitting



Adaptive Multilevel Splitting



Adaptive Multilevel Splitting



Fluctuation Analysis

Fluctuation Analysis

- Thanks to the FK-Del Moral structure, we can carry out martingale analysis an obtain Central Limit Theorem (Warning: practical issues with CLT).

Fluctuation Analysis

- Thanks to the FK-Del Moral structure, we can carry out martingale analysis an obtain Central Limit Theorem (Warning: practical issues with CLT).
- For dynamic AMS, e.g. estimator of rare event is asymptotically normal when $N \to +\infty$ with asymptotic variance:

$$\mathrm{Var}(\hat{p}) = -p^2 \ln p + 2 \int_{s \geqslant 0} \mathrm{Var}_{\eta_s}(\mathbb{P}_{\cdot}(\text{rare event})) d(-p_s^2)$$

Fluctuation Analysis

- Thanks to the FK-Del Moral structure, we can carry out martingale analysis an obtain Central Limit Theorem (Warning: practical issues with CLT).
- For dynamic AMS, e.g. estimator of rare event is asymptotically normal when $N \to +\infty$ with asymptotic variance:

$$\mathrm{Var}(\hat{p}) = -p^2 \ln p + 2 \int_{s \geqslant 0} \mathrm{Var}_{\eta_s}(\mathbb{P}_{\cdot}(\text{rare event})) d(-p_s^2)$$

- Variance is Minimal $= -p^2 \ln p$ for diffusions iff $\mathbb{P}_x$(rare event) only depends on x only through $\xi(x)$.

Fluctuation Analysis

- Thanks to the FK-Del Moral structure, we can carry out martingale analysis an obtain Central Limit Theorem (Warning: practical issues with CLT).
- For dynamic AMS, e.g. estimator of rare event is asymptotically normal when $N \to +\infty$ with asymptotic variance:

$$\mathrm{Var}(\hat{p}) = -p^2 \ln p + 2 \int_{s \geqslant 0} \mathrm{Var}_{\eta_s}(\mathbb{P}_{\cdot}(\text{rare event}))\, d(-p_s^2)$$

- Variance is Minimal $= -p^2 \ln p$ for diffusions iff $\mathbb{P}_x(\text{rare event})$ only depends on x only through $\xi(x)$.
- Ref: Cérou, (Delyon), Guyader , Rousset: series of paper.