

Genealogies of sequential Monte Carlo algorithms

Suzie Brown

27 October 2020

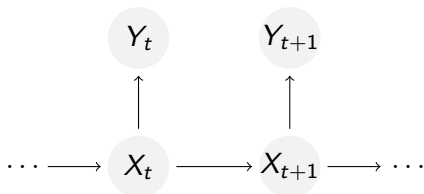
Outline

1. Sequential Monte Carlo
2. Resampling and degeneracy
3. Genealogies

Sequential Monte Carlo

- ▶ Want to sample from a sequence of intractable target distributions
- ▶ Typical settings: dimension of target increases in time, strong dependence between consecutive targets (so MCMC is impractical)
- ▶ SMC can obtain exact draws, and thus approximate expectations

State space models



$$X_0 \sim \mu(\cdot)$$

$$X_{t+1} \mid (X_t = x_t) \sim f_t(\cdot \mid x_t)$$

$$Y_t \mid (X_t = x_t) \sim g_t(\cdot \mid x_t)$$

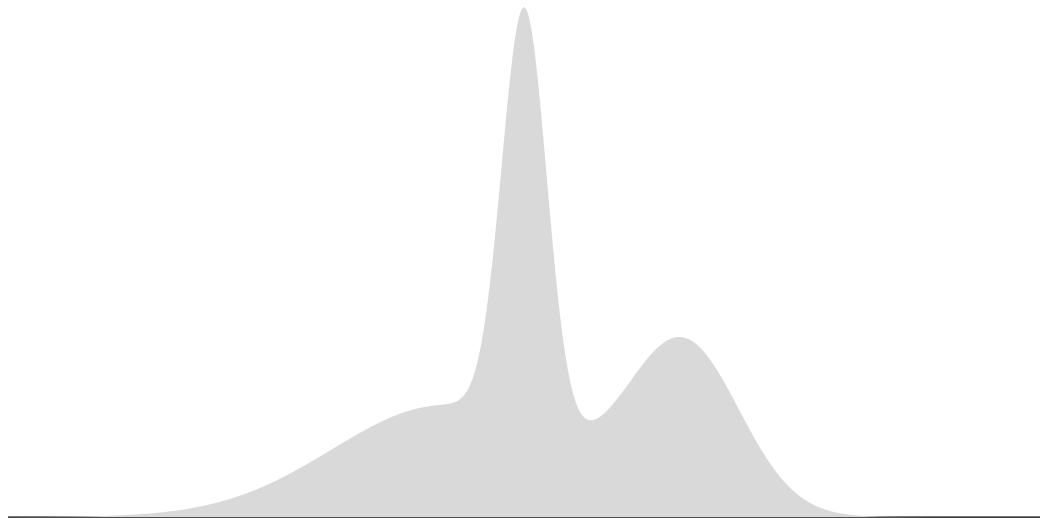
May want to infer ($t < T$):

$$p(x_{1:T} \mid y_{1:t}) \quad \text{"prediction"}$$

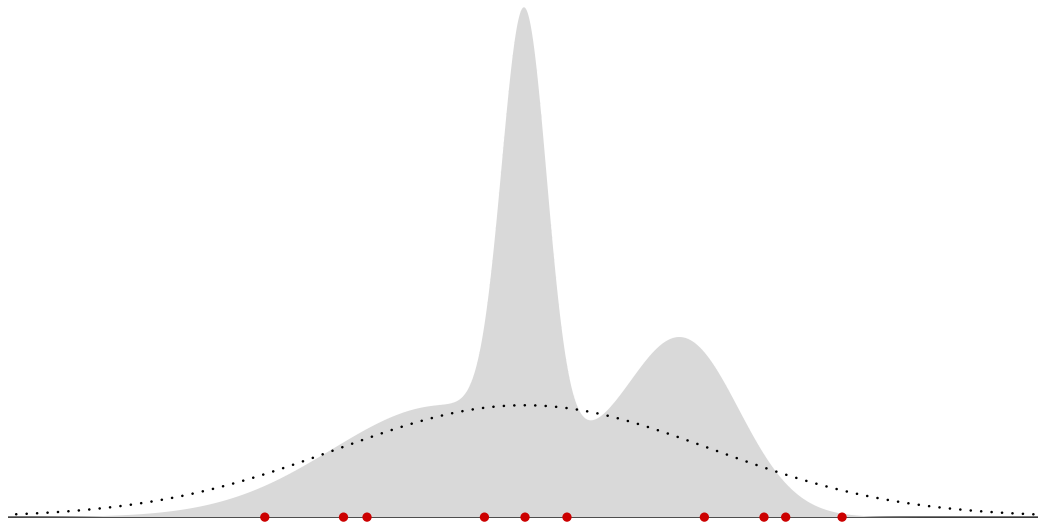
$$p(x_{1:t} \mid y_{1:t}) \quad \text{"filtering"}$$

$$p(x_{1:t} \mid y_{1:T}) \quad \text{"smoothing"}$$

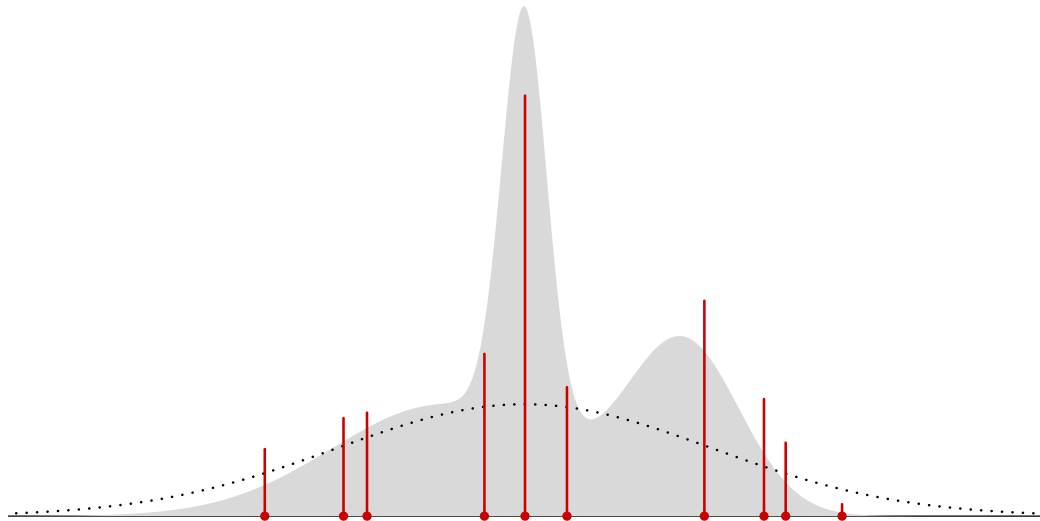
Importance sampling



Importance sampling



Importance sampling

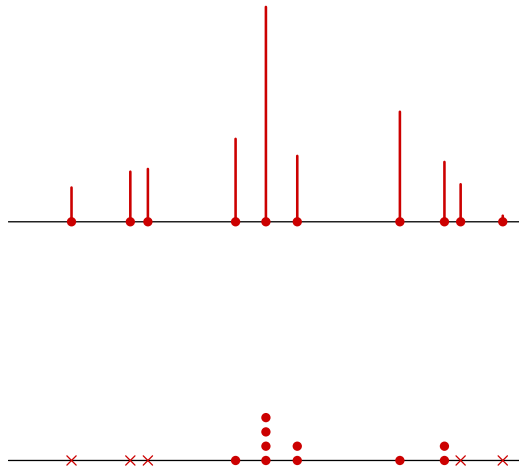


Sequential importance sampling

- ▶ Idea: use weighted samples from one time step to construct a proposal for the next step
- ▶ Multiplying weights over time causes *weight degeneracy*
- ▶ Can avoid this problem by resampling (weights are reset at each step)

Resampling

- ▶ Want to transform continuous weights to discrete offspring counts
- ▶ For example, sample counts from Multinomial distribution

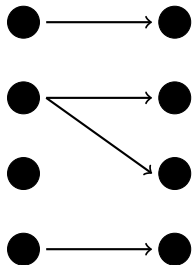


Sequential Monte Carlo algorithm

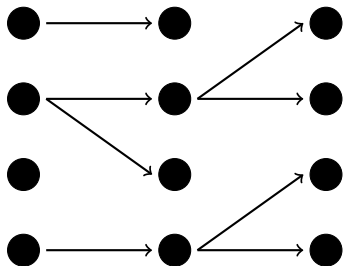
Iterate these steps:

1. Mutate: move particle positions (according to f_t)
2. Weight: Calculate importance weights for each particle
3. Resample: Duplicate/kill particles according to weights, reset weights to $1/N$

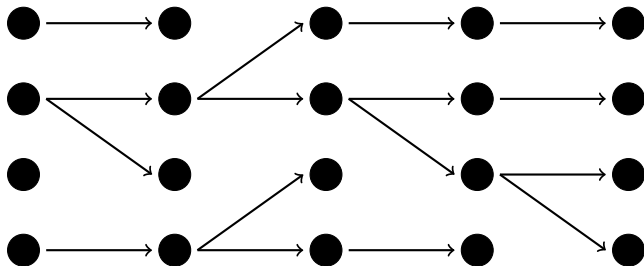
Ancestral degeneracy



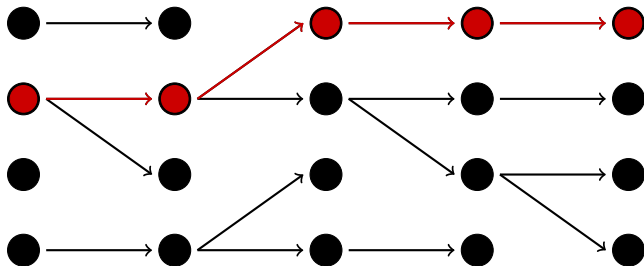
Ancestral degeneracy



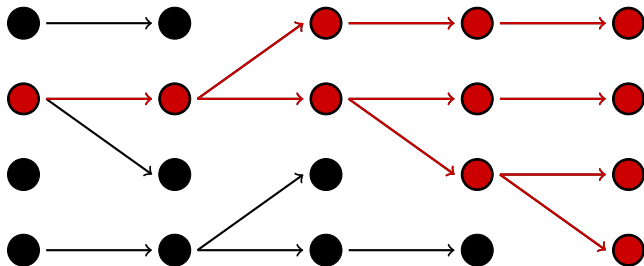
Ancestral degeneracy



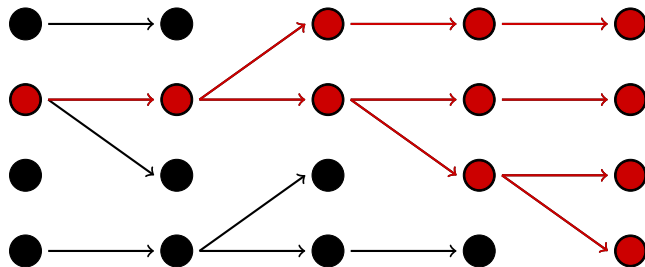
Ancestral degeneracy



Ancestral degeneracy



Ancestral degeneracy

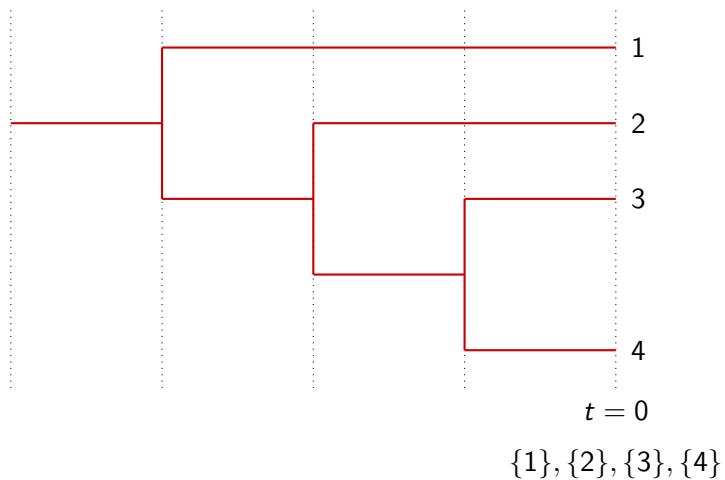


For $t \ll T$, $p(x_{1:t}|y_{1:T})$ is approximated by very few distinct points!

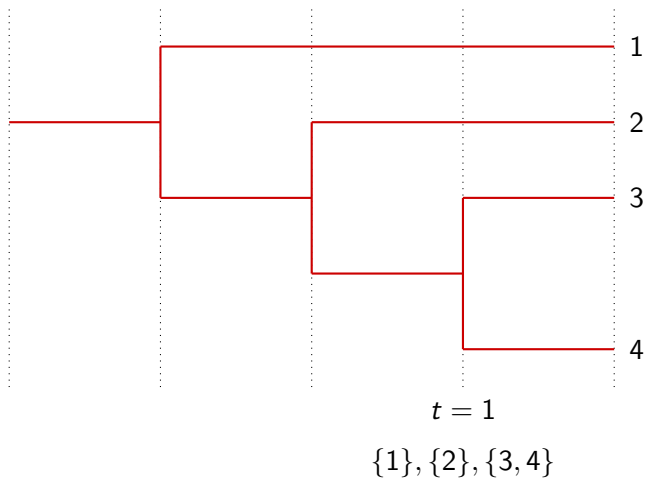
Ancestral degeneracy

- ▶ In smoothing applications, ancestral degeneracy can seriously limit the performance of SMC
- ▶ Can be mitigated by resampling less often (e.g. only when variance of weights gets too big — “adaptive resampling”)
- ▶ Can also use low-variance resampling schemes
- ▶ If we knew how bad ancestral degeneracy would be, we could tune parameters to limit it (e.g. number of particles, threshold for adaptive resampling)
- ▶ Our aim: quantify ancestral degeneracy by analysing the induced genealogy

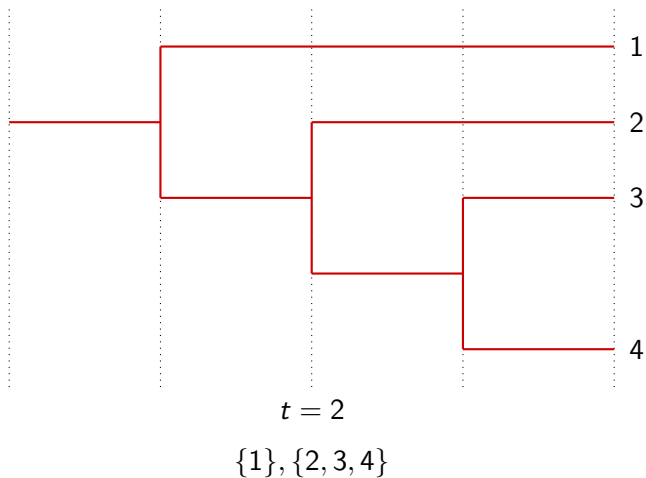
Encoding genealogies



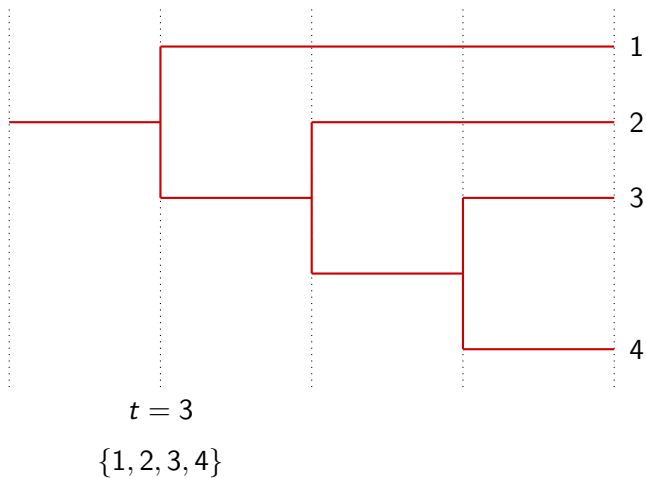
Encoding genealogies



Encoding genealogies



Encoding genealogies

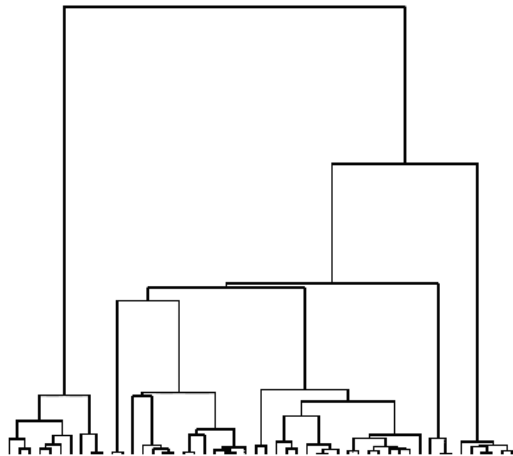


Asymptotic setting

- ▶ We take the number of particles $N \rightarrow \infty$
- ▶ We consider a sample of $n \ll N$ particles at time zero
- ▶ We trace these n lineages back to see the genealogy
- ▶ We will have to rescale time to get a well-defined limiting process

Kingman's n -coalescent

- ▶ Continuous-time Markov chain on the space of equivalence relations on $\{1, \dots, n\}$
- ▶ Only non-identity transitions are those involving a single pair merger
- ▶ This happens at rate 1 for each pair of lineages (total rate $\binom{k}{2}$ while there are k distinct lineages)



Time scale

The probability that a randomly chosen pair of particles at generation t share a common ancestor at generation $(t - 1)$, conditional on offspring counts, is

$$c_N(t) = \frac{1}{N(N-1)} \sum_{i=1}^N v_t^{(i)}(v_t^{(i)} - 1)$$

To get an n -coalescent, this should converge to 1 (the required pair merger rate), so we rescale time by the inverse

$$\tau_N(t) := \min \left\{ s \geq 1 : \sum_{r=1}^s c_N(r) \geq t \right\}$$

Main theorem

- ▶ Offspring counts $(v_t^{(1)}, \dots, v_t^{(N)})$
- ▶ Parent-offspring assignments are uniform given offspring counts
- ▶ Time scale does not explode (i.e. $\mathbb{P}[\tau_N(t) = \infty] = 0$ for all finite t)
- ▶ There exists a sequence (b_N) such that $\lim_{N \rightarrow \infty} b_N = 0$ and

$$\frac{1}{(N)_3} \sum_{i=1}^N \mathbb{E}_t[(v_t^{(i)})_3] \leq b_N \frac{1}{(N)_2} \sum_{i=1}^N \mathbb{E}_t[(v_t^{(i)})_2]$$

Then the finite-dimensional distributions of the rescaled genealogies converge to the n -coalescent as $N \rightarrow \infty$.

Only the method of resampling affects the genealogies of an SMC algorithm. We established the theorem in these cases:

- ▶ Multinomial resampling
- ▶ Stochastic rounding
- ▶ Conditional SMC with multinomial resampling

Multinomial resampling

- ▶ Offspring counts are sampled from Multinomial distribution parametrised by weights
- ▶ Easy to analyse, but doesn't perform well
- ▶ (For the population geneticists): different from Wright-Fisher model because not neutral

Stochastic rounding

- ▶ Minimum-variance resampling schemes
- ▶ Offspring count $v_t^{(i)}$ sampled by rounding $Nw_t^{(i)}$ up or down to an integer
- ▶ Need some way to ensure total number of offspring is N (this is what distinguishes the different schemes within this class)
- ▶ (For the SMC specialists): includes systematic, residual-stratified, residual-systematic, SSP reampling, ...

Conditional SMC

- ▶ Component of particle MCMC
- ▶ One lineage is conditioned to survive all resampling steps
- ▶ Resampling algorithm must deterministically propagate this “immortal lineage”
- ▶ For example: fix one offspring to immortal lineage, multinomial sampling for the remaining counts
- ▶ Conjecture that theorem will also apply to conditional SMC with stochastic rounding

In conclusion...

- ▶ Genealogies can help us to analyse performance of smoothing algorithms which suffer ancestral degeneracy
- ▶ We have simple conditions under which these genealogies converge to n -coalescent
- ▶ These conditions are verified for some important classes of SMC algorithms

In conclusion...

- ▶ Genealogies can help us to analyse performance of smoothing algorithms which suffer ancestral degeneracy
- ▶ We have simple conditions under which these genealogies converge to n -coalescent
- ▶ These conditions are verified for some important classes of SMC algorithms

Open questions

- ▶ How to estimate the time scale c_N a priori (since it depends on offspring counts)
- ▶ Weak convergence (as opposed to FDDs) so we can say more about convergence of expectations
- ▶ Rate of convergence
- ▶ Finite- N behaviour

THE END

For more details you can look at the pre-print
[arXiv:2007.00096](https://arxiv.org/abs/2007.00096)