Memory (and Time) Efficient SMC

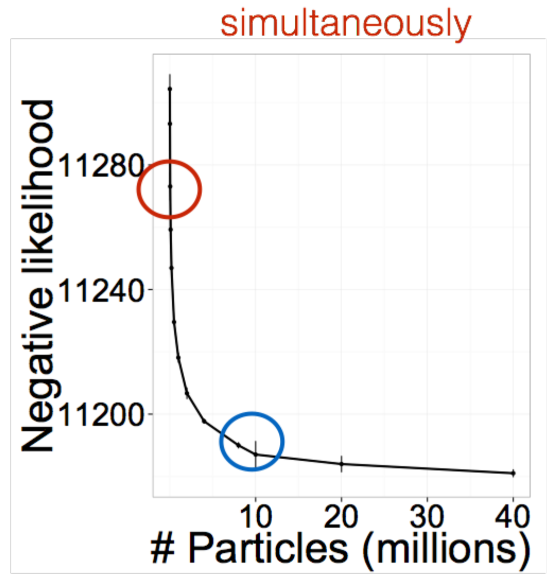
Seong-Hwan Jun and Alexandre Bouchard-Côté
Department of Statistics, The University of British Columbia

Contribution: Implicit particle SMC

- Designed a new SMC algorithm that can use more particles (N) than can fit in the memory (K)
- Main idea: replay randomness

Motivation: SMC for Phylogenetics

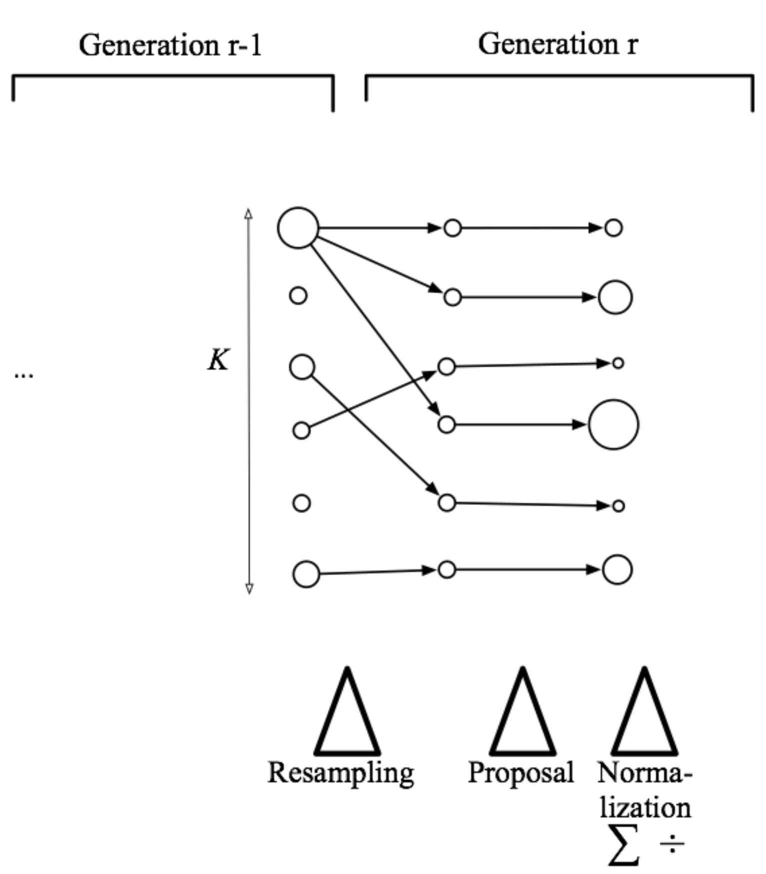
Memory per particle: ~100KB for moderate sized data.
Problem: need to store many particles simultaneously



- Elbow at around 10 million particles
- Can only fit approx 40,000 particles in a standard computer (8GB RAM)
- Our method allows to go all the way to 40 million particles (under same constraint)

Experiments on:
-20 observed leaves
-1000 sites

Background: SMC



K is a constant over generations

- Resampling:
  $\tilde{x}_{r-1,k} \sim \text{Mult}(\bar{w}_{r-1,1}, \dots, \bar{w}_{r-1,K})$
- Proposal:
  $x_{r,k} \sim q(\cdot|\tilde{x}_{r-1,k})$
- Weight computation:
  $w_{r,k} := w(\tilde{x}_{r-1,k} \to x_{r,k})$
- Normalization (summation):
  $s \leftarrow s + w_{r,k}$
- Normalization (division):
  $s \leftarrow w_{r,k}/s$

$$\hat{\pi}_{r,K}(x) = \sum_{k=1}^{K} \bar{w}_{r,k}\delta_{x_{r,k}}(x)$$

$$\int \phi d\hat{\pi}_{r,K} \xrightarrow{L^2} \int \phi d\pi_r \text{ as } K \to \infty$$

Replaying randomness

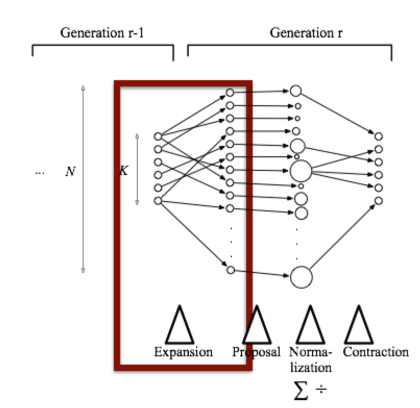
Stochastic map

$x \sim q(\cdot|\tilde{x})$ ⟷ $x = q(U, \tilde{x})$, $U \sim \text{Unif}(0,1)$

- Realization of a stochastic map: $q_u(x)$
- In practice, only need the seed (O(1) memory)
- Can replay both the resampling and the proposal steps

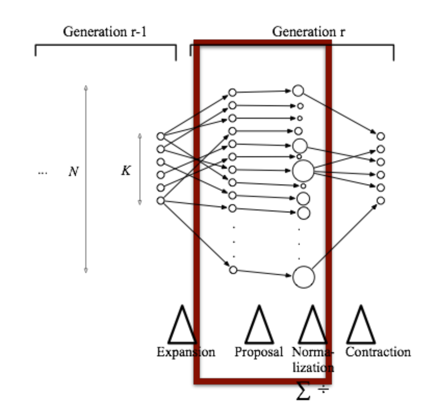
IPSMC: overview

1. Expansion



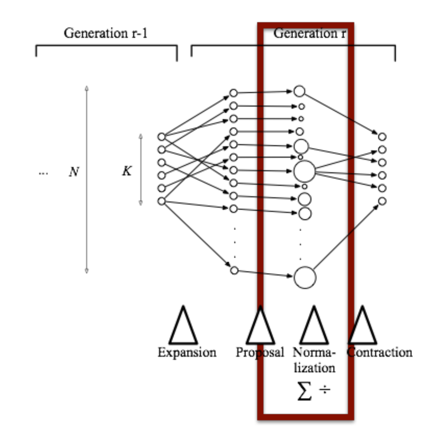
$\tilde{X}_{r-1,1}, \dots, \tilde{X}_{r-1,N} \sim \text{Mult}(\bar{w}_{r-1,1}, \dots, \bar{w}_{r-1,K})$

2. Implicit proposal



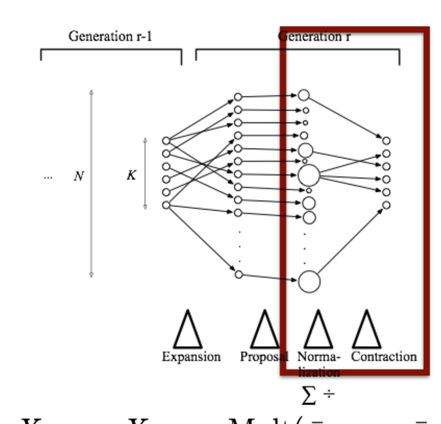
$x_{r,k} \sim q_u(\cdot|\tilde{x}_{r-1,k})$

3. Normalization



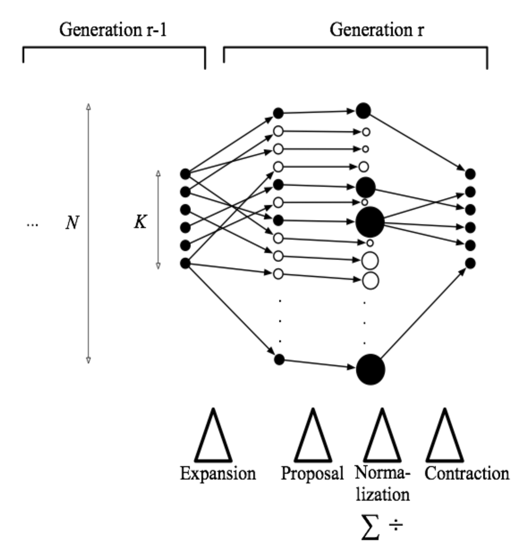
$s \leftarrow s + w_{r,k} \quad \bar{w}_{r,k} \leftarrow w_{r,k}/s$

4. Contraction

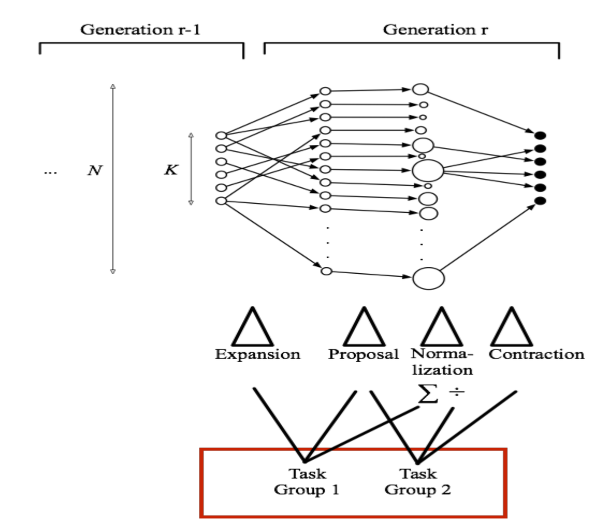


$X_{r,1}, \dots, X_{r,K} \sim \text{Mult}(\bar{w}_{r,1}, \dots, \bar{w}_{r,N})$

5. Concrete proposal



Streaming particle IPSMC



Adaptive number of particles

Proposition 1: Let $X_1, \dots, X_K \sim \text{Mult}(\bar{w})$ independently, where $\bar{w} = (\bar{w}_1, \dots, \bar{w}_N)$. Then, we have

$$\psi(w, K) = \mathbb{E}|\{X_1, \dots, X_K\}| = N - \sum_{i=1}^{N}(1 - \bar{w}_i)^K$$

- $N(K, M) = \sup\{n \le N^* : \psi(w, K) \le M\}$; N^* is a computational ceiling
- M is a tuning parameter

Proposition 2: Let S be the number of distinct particles sampled from a multinomial distribution. Then, $Var(S) \le 3K$ whenever $\max_i \bar{w}_i < 1/2$. By Chebyshev's inequality, for $\epsilon > 0$:

$$\mathbb{P}(|S - \mathbb{E}S| \ge \epsilon \mathbb{E}S) \le \frac{3K}{\epsilon^2(\mathbb{E}S)^2}$$

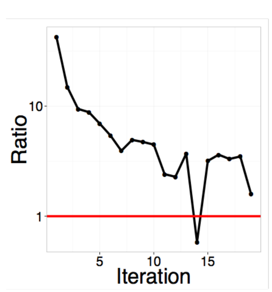
Proposition 3 (consistency): Assume that the test function ϕ and the un-normalized weights w are bounded, and that the proposal distributions ν satisfy $\pi_r << \nu_{r-1,x}$. Then,

$$\int \phi d\pi_{r,K}^{IP} \xrightarrow{L^2} \int \phi d\pi_r$$

where $\pi_{r,K}^{IP} = \text{res}_K\left(\text{prop}_{N(K,M)}\,\pi_{r-1,K}^{IP}\right)$. Here res is an operator that corresponds to the contraction step and prop is an operator corresponding to expansion and implicit proposal steps.

Time efficiency

Phylogenetics speed-up results

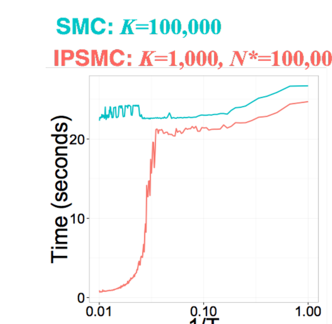


- Computed the ratio of the *effective sample size* per seconds (ESS/s) for IPSMC to the ESS/s for SMC

$$ess = \frac{1}{\sum_{k=1}^{K} \bar{w}_k^2}$$

- Interested in the quality of samples generated per unit of computation time

Ising Model



- Problem gets difficult as T gets smaller
- Experiment on 32x32 lattice
- Compare IPSMC to SMC both given 1GB of RAM
- Both methods give comparable approximations of the true marginals

Conclusion and Other applications

Designed a new SMC algorithm that can use more particles than can fit in memory.

- Probabilistic programming contexts (where proposal could be arbitrarily poor)
- State space models where transition dynamics cannot be evaluated point wise