

Divide-and-Conquer Sequential Monte Carlo

Adam M. Johansen

Joint work with:

John Aston, Alexandre Bouchard-Côté, Brent Kirkpatrick,
Fredrik Lindsten, Christian Næseth, Thomas Schön

University of Warwick

a.m.johansen@warwick.ac.uk

www2.warwick.ac.uk/fac/sci/statistics/staff/academic/johansen/talks/

OxWaSP

OxWaSP/Amazon Meeting, Berlin
May 12th, 2016

Outline

- ▶ Sequential Monte Carlo
- ▶ Divide and Conquer SMC
- ▶ Illustrative Applications

Essential Problem

The Abstract Problem

- ▶ Given a distribution,

$$\pi(x) = \frac{\gamma(x)}{Z},$$

- ▶ such that $\gamma(x)$ can be evaluated pointwise,
- ▶ how can we approximate π
- ▶ and how about Z ?
- ▶ Can we do so robustly?
- ▶ In a distributed setting?

Importance Sampling

- ▶ Simple Identity, provided $\gamma \ll \mu$:

$$Z = \int \gamma(x) dx = \int \frac{\gamma(x)}{\mu(x)} \mu(x) dx$$

- ▶ So, if $X_1, X_2, \dots \stackrel{\text{iid}}{\sim} \mu$, then:

unbiasedness

$$\forall N : \mathbb{E} \left[\frac{1}{N} \sum_{i=1}^N \frac{\gamma(X_i)}{\mu(X_i)} \right] = Z$$

slln

$$\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{i=1}^N \frac{\gamma(X_i)}{\mu(X_i)} \varphi(X_i) \xrightarrow{\text{a.s.}} \gamma(\varphi)$$

clt

$$\lim_{N \rightarrow \infty} \sqrt{N} \left[\frac{1}{N} \sum_{i=1}^N \frac{\gamma(X_i)}{\mu(X_i)} \varphi(X_i) - \gamma(\varphi) \right] \xrightarrow{d} W$$

where $W \sim \mathcal{N} \left(0, \text{Var} \left[\frac{\gamma(X_1)}{\mu(X_1)} \varphi(X_1) \right] \right)$.

Sequential Importance Sampling

- ▶ Write

$$\gamma(x_{1:n}) = \gamma(x_1) \prod_{p=2}^n \gamma(x_p | x_{1:p-1}),$$

- ▶ define, for $p = 1, \dots, n$

$$\gamma_p(x_{1:p}) = \gamma_1(x_1) \prod_{q=2}^p \gamma(x_q | x_{1:q-1}),$$

- ▶ then

$$\underbrace{\frac{\gamma(x_{1:n})}{\mu(x_{1:n})}}_{W_n(x_{1:n})} = \underbrace{\frac{\gamma_1(x_1)}{\mu_1(x_1)}}_{w_1(x_1)} \prod_{p=2}^n \underbrace{\frac{\gamma_p(x_{1:p})}{\gamma_{p-1}(x_{1:p-1})\mu_p(x_p | x_{1:p-1})}}_{w_p(x_{1:p})},$$

- ▶ and we can *sequentially* approximate $Z_p = \int \gamma_p(x_{1:p}) dx_{1:p}$.

Sequential Importance Resampling (SIR)

Given a sequence $\gamma_1(x_1), \gamma_2(x_{1:2}), \dots$:

Initialisation, $n = 1$:

- ▶ Sample $X_1^1, \dots, X_1^N \stackrel{\text{iid}}{\sim} \mu_1$

- ▶ Compute

$$W_1^i = \frac{\gamma_1(X_1^i)}{\mu_1^i(X_1^i)}$$

- ▶ Obtain $\hat{Z}_1^N = \frac{1}{N} \sum_{i=1}^N W_1^i$

[This is *just* importance sampling.]

Iteration, $n \leftarrow n + 1$:

- ▶ Resample: sample $(X_{n,1:n-1}^1, \dots, X_{n,1:n-1}^N) \stackrel{\text{iid}}{\sim} \sum_{i=1}^N \delta_{X_{n-1}^i}$
- ▶ Sample $X_{n,n}^i \sim q_n(\cdot | X_{n,1:n-1}^i)$
- ▶ Compute

$$W_n^i = \frac{\gamma_n(X_{n,1:n}^i)}{\gamma_{n-1}(X_{n,1:n-1}^i) \cdot q_n(X_{n,n}^i | X_{n,1:n-1}^i)}.$$

- ▶ Obtain

$$\hat{Z}_n^N = \hat{Z}_{n-1}^N \cdot \frac{1}{N} \sum_{i=1}^N W_n^i.$$

SIR: Theoretical Justification

Under regularity conditions we still have:

unbiasedness

$$\mathbb{E}[\hat{Z}_n^N] = Z_n$$

slln

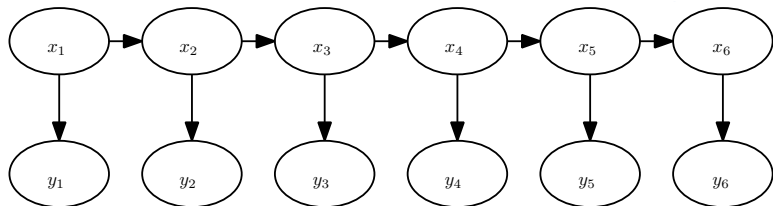
$$\lim_{N \rightarrow \infty} \hat{\pi}_n^N(\varphi) \stackrel{\text{a.s.}}{=} \pi_n(\varphi)$$

clt For a normal random variable W_n of appropriate variance:

$$\lim_{N \rightarrow \infty} \sqrt{N}[\hat{\pi}_n^N(\varphi) - \pi_n(\varphi)] \stackrel{d}{=} W_n$$

although establishing this becomes a little harder (cf., e.g. Del Moral (2004), Andrieu et al. 2010).

Simple Particle Filters: One Family of SIR Algorithms



- ▶ Unobserved Markov chain $\{X_n\}$ transition f .
- ▶ Observed process $\{Y_n\}$ conditional density g .
- ▶ The joint density is available:

$$p(x_{1:n}, y_{1:n}|\theta) = f_1^\theta(x_1)g^\theta(y_1|x_1) \prod_{i=2}^n f^\theta(x_i|x_{i-1})g^\theta(y_i|x_i).$$

- ▶ Natural SIR target distributions:

$$\pi_n^\theta(x_{1:n}) := p(x_{1:n}|y_{1:n}, \theta) \propto p(x_{1:n}, y_{1:n}|\theta) =: \gamma_n^\theta(x_{1:n})$$

$$Z_n^\theta = \int p(x_{1:n}, y_{1:n}|\theta) dx_{1:n} = p(y_{1:n}|\theta)$$

Bootstrap PFs and Similar

- ▶ Choosing

$$\pi_n^\theta(x_{1:n}) := p(x_{1:n}|y_{1:n}, \theta) \propto p(x_{1:n}, y_{1:n}|\theta) =: \gamma_n^\theta(x_{1:n})$$

$$Z_n^\theta = \int p(x_{1:n}, y_{1:n}|\theta) dx_{1:n} = p(y_{1:n}|\theta)$$

- ▶ and $q_p(x_p|x_{1:p-1}) = f^\theta(x_p|x_{p-1})$ yields the bootstrap particle filter of Gordon et al. (1993),
- ▶ whereas $q_p(x_p|x_{1:p-1}) = p(x_p|x_{p-1}, y_p, \theta)$ yields the “locally optimal” particle filter.
- ▶ Note: Many alternative particle filters are SIR algorithms with other targets. Cf. J. and Doucet (2008); Doucet and J. (2011).

Sequential Monte Carlo Samplers: Another SIR Class

Given a sequence of targets π_1, \dots, π_n on *arbitrary* spaces, Del Moral et al. (2006) extend the space:

$$\tilde{\pi}_n(x_{1:n}) = \pi_n(x_n) \prod_{p=n-1}^1 L_p(x_{p+1}, x_p)$$

$$\tilde{\gamma}_n(x_{1:n}) = \gamma_n(x_n) \prod_{p=n-1}^1 L_p(x_{p+1}, x_p)$$

$$\begin{aligned} \tilde{Z}_n &= \int \tilde{\gamma}_n(x_{1:n}) dx_{1:n} \\ &= \int \gamma_n(x_n) \prod_{p=n-1}^1 L_p(x_{p+1}, x_p) dx_{1:n} = \int \gamma_n(x_n) dx_n = Z_n \end{aligned}$$

A Simple SMC Sampler

Given $\gamma_1, \dots, \gamma_n$, on (E, \mathcal{E}) , for $i = 1, \dots, N$

- ▶ Sample $X_1^i \stackrel{\text{iid}}{\sim} \mu_1$ compute $W_1^i = \frac{\gamma_1(X_1^i)}{\mu_1(X_1^i)}$ and $\hat{Z}_1^N = \frac{1}{N} \sum_{i=1}^N W_1^i$
- ▶ For $p = 2, \dots, n$
 - ▶ Resample: $X_{n,n-1}^{1:N} \stackrel{\text{iid}}{\sim} \sum_{i=1}^N W_{n-1}^i \delta_{X_{n-1}^i}$.
 - ▶ Sample: $X_n^i \sim K_n(X_{n,1:n-1}^i, \cdot)$, where $\pi_n K_n = \pi_n$.
 - ▶ Compute: $W_n^i = \frac{\gamma_n(X_{n,n-1}^i)}{\gamma_{n-1}(X_{n,n-1}^i)}$.
 - ▶ And $\hat{Z}_n^N = \hat{Z}_{n-1}^N \cdot \frac{1}{N} \sum_{i=1}^N W_n^i$.

Bayesian Inference

(Chopin, 2001; Del Moral et al., 2006)

In a Bayesian context:

- ▶ Given a prior $p(\theta)$ and likelihood $l(\theta; y_{1:m})$
- ▶ One could specify:

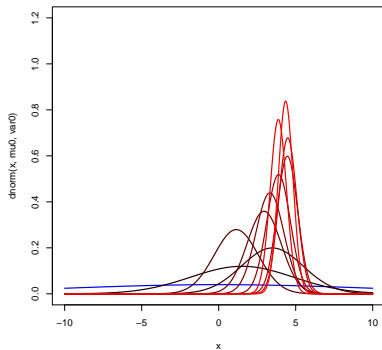
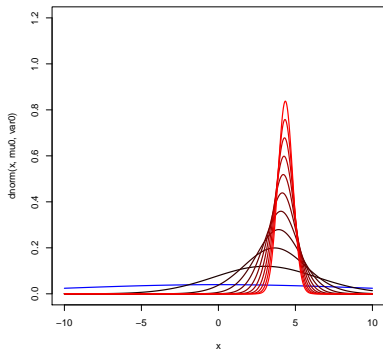
Data Tempering $\gamma_p(\theta) = p(\theta)l(\theta; y_{1:m_p})$ for
 $m_1 = 0 < m_2 < \dots < m_T = m$

Likelihood Tempering $\gamma_p(\theta) = p(\theta)l(\theta; y_{1:m})^{\beta_p}$ for
 $\beta_1 = 0 < \beta_2 < \dots < \beta_T = 1$

Something else?

- ▶ Here $Z_T = \int p(\theta)l(\theta; y_{1:n})d\theta$ and $\gamma_T(\theta) \propto p(\theta|y_{1:n})$.
- ▶ Specifying (m_1, \dots, m_T) , $(\beta_1, \dots, \beta_T)$ or $(\gamma_1, \dots, \gamma_T)$ is hard.

Illustrative Sequences of Targets



One Adaptive Scheme (Zhou, J. & Aston, 2016)+Refs

Resample When ESS is below a threshold.

Likelihood Tempering At iteration n : Set α_n such that:

$$\frac{N(\sum_{j=1}^N W_{n-1}^{(j)} w_n^{(j)})^2}{\sum_{k=1}^N W_{n-1}^{(k)} (w_n^{(k)})^2} = \text{CESS}_*$$

Proposals Follow (Jasra et al., 2010): adapt to keep acceptance rate about right.

Question

Are there better, practical approaches to specifying a sequence of distributions?

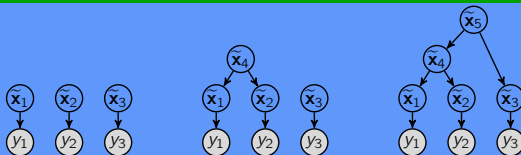
Divide-and-Conquer (Lindsten, J. et al., 2014)

Many models admit natural decompositions:

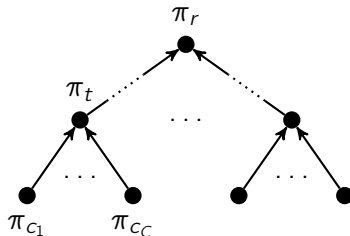
Level 0:

Level 1:

Level 2:

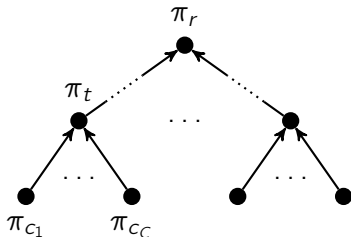


To which we can apply a divide-and-conquer strategy:



A few formalities...

- ▶ Use a tree, T of models (with rootward variable inclusion):



- ▶ Let $t \in T$ denote a node; $r \in T$ is the root.
- ▶ Let $\mathcal{C}(t) = \{c_1, \dots, c_C\}$ denote the children of t .
- ▶ Let $\tilde{\mathcal{X}}_t$ denote the space of variables included in t but *not* its children.
- ▶ dc-smc can be viewed as a recursion over this tree.

dc-smc(t) an extension of SIR

1. For $c \in \mathcal{C}(t)$:
 - 1.1 $(\{X_c^i, W_c^i\}_{i=1}^N, \widehat{Z}_c^N) \leftarrow \text{dc-smc}(c)$.
 - 1.2 Resample $\{\mathbf{x}_c^i, \mathbf{w}_c^i\}_{i=1}^N$ to obtain the equally weighted particle system $\{\hat{\mathbf{x}}_c^i, 1\}_{i=1}^N$.
2. For particle $i = 1 : N$:
 - 2.1 If $\tilde{\mathcal{X}}_t \neq \emptyset$, simulate $\tilde{\mathbf{x}}_t^i \sim q_t(\cdot \mid \hat{\mathbf{x}}_{c_1}^i, \dots, \hat{\mathbf{x}}_{c_C}^i)$, where $(c_1, c_2, \dots, c_C) = \mathcal{C}(t)$;
else $\tilde{\mathbf{x}}_t^i \leftarrow \emptyset$.
 - 2.2 Set $\mathbf{x}_t^i = (\hat{\mathbf{x}}_{c_1}^i, \dots, \hat{\mathbf{x}}_{c_C}^i, \tilde{\mathbf{x}}_t^i)$.
 - 2.3 Compute $\mathbf{w}_t^i = \frac{\gamma_t(\mathbf{x}_t^i)}{\prod_{c \in \mathcal{C}(t)} \gamma_c(\hat{\mathbf{x}}_c^i)} \frac{1}{q_t(\tilde{\mathbf{x}}_t^i \mid \hat{\mathbf{x}}_{c_1}^i, \dots, \hat{\mathbf{x}}_{c_C}^i)}$.
3. Compute $\widehat{Z}_t^N = \left\{ \frac{1}{N} \sum_{i=1}^N \mathbf{w}_t^i \right\} \prod_{c \in \mathcal{C}(t)} \widehat{Z}_c^N$.
4. Return $(\{\mathbf{x}_t^i, \mathbf{w}_t^i\}_{i=1}^N, \widehat{Z}_t^N)$.

Theoretical Properties I

Unbiasedness of Normalising Constant Estimates

Provided that $\gamma_t \ll \otimes_{c \in \mathcal{C}(t)} \gamma_c \otimes q_t$ for every $t \in \mathcal{T}$ and an unbiased, exchangeable resampling scheme is applied to every population at every iteration, we have for any $N \geq 1$:

$$\mathbb{E}[\hat{Z}_r^N] = Z_r = \int \gamma_r(\mathbf{x}_r) d\mathbf{x}_r.$$

Strong Law of Large Numbers

Under regularity conditions the weighted particle system $(\mathbf{x}_{r,N}^i, \mathbf{w}_{r,N}^i)_{i=1}^N$ generated by $\text{dc-smc}(r)$ is consistent in that for all functions $f : \mathcal{Z} \rightarrow \mathbb{R}$ satisfying certain assumptions:

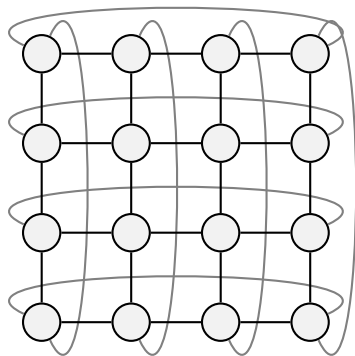
$$\sum_{i=1}^N \frac{\mathbf{w}_r^{N,i}}{\sum_{j=1}^N \mathbf{w}_r^{N,j}} f(\mathbf{x}_{r,N}^i) \xrightarrow{\text{a.s.}} \int f(\mathbf{x}) \pi(\mathbf{x}) d\mathbf{x}, \quad \text{as } N \rightarrow \infty.$$

Some (Importance) Extensions

1. Mixture Resampling
2. Tempering (Del Moral et al, 2006)
3. Adaptation (Zhou, J. and Aston, 2016)

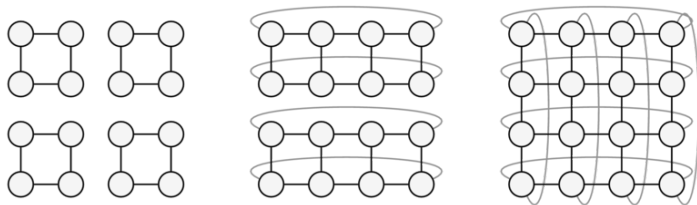
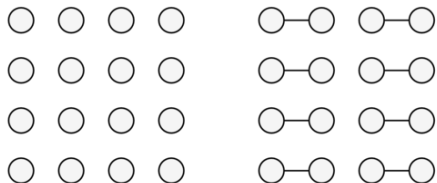
An Ising Model

- ▶ k indexes $\mathcal{V} \in \mathcal{V} \subset \mathbb{Z}^2$
- ▶ $x_k \in \{-1, 1\}$
- ▶ $p(\mathbf{z}) \propto e^{-\beta E(\mathbf{z})}$, $\beta \geq 0$
- ▶ $E(\mathbf{z}) = -\sum_{(k,l) \in \mathcal{E}} x_k x_l$

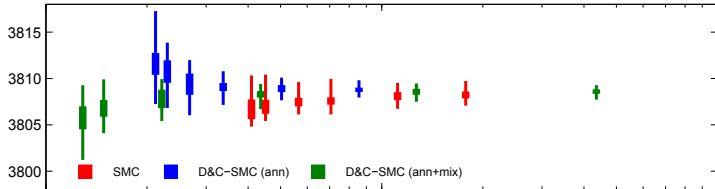


We consider a grid of size 64×64 with $\beta = 0.4407$ (the critical temperature).

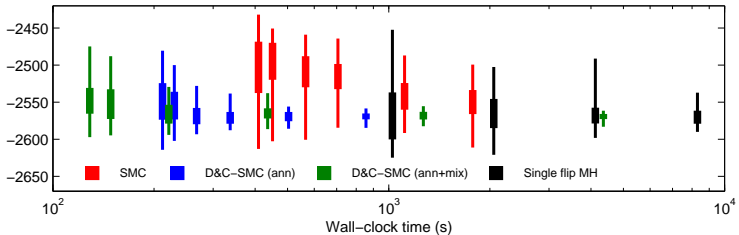
A sequence of decompositions



$\log Z$



$\mathbb{E}[E(\mathbf{x})]$



Summaries over 50 independent runs of each algorithm.

New York Schools Maths Test: data

- ▶ Data organised into a tree T .
- ▶ A root-to-leaf path is: NYC (the root, denoted by $r \in T$), borough, school district, school, year.
- ▶ Each leaf $t \in T$ comes with an observation of m_t exam successes out of M_t trials.
- ▶ Total of 278 399 test instances
- ▶ five borough (Manhattan, The Bronx, Brooklyn, Queens, Staten Island),
- ▶ 32 distinct districts,
- ▶ 710 distinct schools.

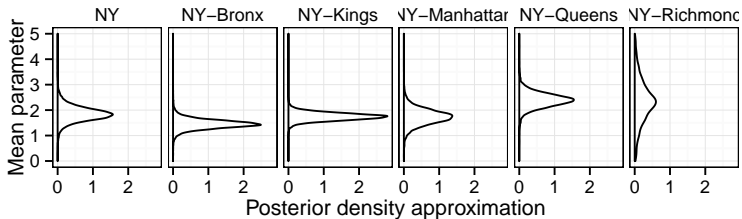
New York Schools Maths Test: Bayesian Model

- ▶ Number of successes m_t at a leaf t is $\text{Bin}(M_t, p_t)$.
- ▶ where $p_t = \text{logistic}(\theta_t)$, where θ_t is a latent parameter.
- ▶ internal nodes of the tree also have a latent θ_t
- ▶ model the difference in θ_t along $e = (t \rightarrow t')$ as
$$\theta_{t'} = \theta_t + \Delta_e,$$
- ▶ where, $\Delta_e \sim \text{N}(0, \sigma_e^2)$.
- ▶ We put an improper prior (uniform on $(-\infty, \infty)$) on θ_r .
- ▶ We also make the variance random, but shared across siblings, $\sigma_t^2 \sim \text{Exp}(1)$.

New York Schools Maths Test: Implementation

- ▶ The basic SIR-implementation of dc-smc.
- ▶ Using the natural hierarchical structure provided by the model.
- ▶ Given σ_t^2 and the θ_t at the leaves, the other random variables are multivariate normal.
- ▶ We instantiate values for θ_t only at the leaves.
- ▶ At internal node t' , sample only $\sigma_{t'}^2$ and marginalize out $\theta_{t'}$.
- ▶ Each step of dc-smc therefore is either:
 - At leaves sample $p_t \sim \text{Beta}(1 + m_t, 1 + M_t - m_t)$ and set $\theta_t = \text{logit}(p_t)$.
 - At internal nodes sample $\sigma_t^2 \sim \text{Exp}(1)$.
- ▶ Java implementation:
<https://github.com/alexandrebourchard/multilevelSMC>

New York Schools Maths Test: Results

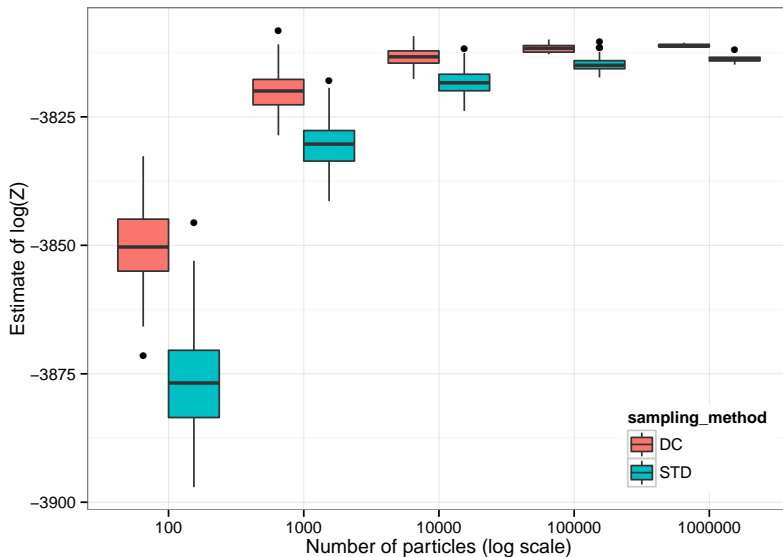


- ▶ DC with 10 000 particles.
- ▶ Bronx County has the highest fraction (41%) of children (under 18) living below poverty level.¹
- ▶ Queens has the second lowest (19.7%),
- ▶ after Richmond (Staten Island, 16.7%).
- ▶ Staten Island contains a single school district so the posterior distribution is much flatter for this borough.

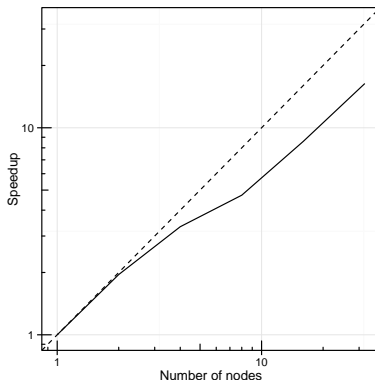
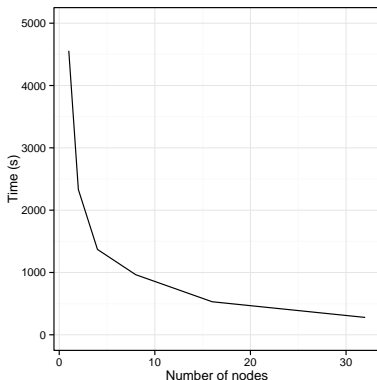
¹Statistics from the New York State Poverty Report 2013,

http://ams.nyscommunityaction.org/Resources/Documents/News/NYSKAAs_2013_Poverty_Report.pdf

Normalising Constant Estimates



Distributed Implementation



Xeon X5650 2.66GHz processors connected by a non-blocking Infiniband 4X QDR network

Conclusions

- ▶ $\text{SMC} \approx \text{SIR}$
- ▶ $\text{D\&C-SMC} \approx \text{SIR} + \text{Coalescence}$
- ▶ Distributed implementation is straightforward
- ▶ D&C strategy can improve even serial performance
- ▶ Some questions remain unanswered:
 - ▶ How can we construct (near) optimal tree-decompositions?
 - ▶ How much standard SMC theory can be extended to this setting?

References

- [1] C. Andrieu, A. Doucet, and R. Holenstein. Particle Markov chain Monte Carlo. *Journal of the Royal Statistical Society B*, 72(3):269–342, 2010.
- [2] N. Chopin. A sequential particle filter method for static models. *Biometrika*, 89(3):539–551, 2002.
- [3] P. Del Moral. *Feynman-Kac formulae: genealogical and interacting particle systems with applications*. Probability and Its Applications. Springer Verlag, New York, 2004.
- [4] P. Del Moral, A. Doucet, and A. Jasra. Sequential Monte Carlo methods for Bayesian Computation. In *Bayesian Statistics 8*. Oxford University Press, 2006.
- [5] A. Doucet and A. M. Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. In D. Crisan and B. Rozovsky, editors, *The Oxford Handbook of Nonlinear Filtering*, pages 656–704. Oxford University Press, 2011.
- [6] N. J. Gordon, S. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings-F*, 140(2):107–113, April 1993.
- [7] A. Jasra, D. A. Stephens, A. Doucet, and T. Tsagaris. Inference for Lévy-Driven Stochastic Volatility Models via Adaptive Sequential Monte Carlo. *Scandinavian Journal of Statistics*, 38(1):1–22, Dec. 2010.
- [8] A. M. Johansen and A. Doucet. A note on the auxiliary particle filter. *Statistics and Probability Letters*, 78(12):1498–1504, September 2008. URL <http://dx.doi.org/10.1016/j.spl.2008.01.032>.
- [9] F. Lindsten, A. M. Johansen, C. A. Næseth, B. Kirkpatrick, T. Schön, J. A. D. Aston, and A. Bouchard-Côté. Divide and conquer with sequential Monte Carlo samplers. Technical Report 1406.4993, ArXiv Mathematics e-prints, 2014.
- [10] Y. Zhou, A. M. Johansen, and J. A. D. Aston. Towards automatic model comparison: An adaptive sequential Monte Carlo approach. *Journal of Computational and Graphical Statistics*, 2015. In press.