Sequential Monte Carlo for graphical models: Graph decompositions and Divide-and-Conquer SMC

Fredrik Lindsten
Department of Engineering
The University of Cambridge
Cambridge, UK

Ongoing joint work with: Christian A. Naesseth (Linköping University), Thomas Schön (Uppsala University), Adam M. Johansen (University of Warwick), Bonnie Kirkpatrick (University of Miami), John Aston (University of Cambridge) and Alexandre Bouchard-Côté (University of British Columbia).
A **probabilistic graphical model** (PGM) is a probabilistic model where a graph \( \mathcal{G} = (\mathcal{V}, \mathcal{E}) \) represents the conditional independency structure between random variables,

1. a set of **vertices** \( \mathcal{V} \) (nodes) represents the random variables
2. a set of **edges** \( \mathcal{E} \) containing elements \((i, j) \in \mathcal{E} \) connecting a pair of nodes \((i, j) \in \mathcal{V} \times \mathcal{V} \)

\[
p(x_{0:T}, y_{1:T}) = p(x_0) \prod_{t=1}^{T} p(x_t | x_{t-1}) \prod_{t=1}^{T} p(y_t | x_t).
\]

Fredrik Lindsten, *Sequential Monte Carlo for graphical models*
Statistics Seminar, University of Bristol, 7 November 2014.
For an undirected graphical model (Markov random field), the joint PDF over all the involved random variables $X_{\mathcal{V}} := (x_i)_{i \in \mathcal{V}}$ is

$$p(X_{\mathcal{V}}) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(X_C),$$

where $\mathcal{C}$ is the set of cliques in $G$, and $Z = \int \prod_{C \in \mathcal{C}} \psi_C(X_C) dX_{\mathcal{V}}$.

Example of a **factor graph** making interactions explicit,

$$p(x_{1:5}) = \frac{1}{Z} \prod_{i=1}^{5} \psi_i(\cdot).$$
Approximate a sequence of probability distributions on a sequence of probability spaces of increasing dimension.

Let \( \{ \gamma_k(x_{1:k}) \}_{k \geq 1} \) be a sequence of unnormalised densities and

\[
\bar{\gamma}_k(x_{1:k}) = \frac{\gamma_k(x_{1:k})}{Z_k}
\]

Approximates

\[
\bar{\gamma}_k(x_{1:k}) \approx \sum_{i=1}^{N} \frac{w_i}{\sum_{l=1}^{N} w_l} \delta_{x^{i}_{1:k}}(x_{1:k}).
\]

**Ex. (SSM)**

\[
\bar{\gamma}_k(x_{1:k}) = p(x_{1:k} \mid y_{1:k}), \quad \gamma_k(x_{1:k}) = p(x_{1:k}, y_{1:k}),
\]

\[
Z_k = p(y_{1:k}).
\]
1. **Resampling**: \( \{ x_{1:k-1}^i, w_{k-1}^i \}_{i=1}^N \rightarrow \{ \tilde{x}_{1:k-1}^i, 1 \}_{i=1}^N \).

2. **Propagation**: \( x_k^i \sim q_k(x_k | \tilde{x}_{1:k-1}^i) \) and \( x_{1:k}^i = \{ \tilde{x}_{1:k-1}^i, x_k^i \} \).

3. **Weighting**: \( w_k^i = W_k(x_{1:k}^i) = \frac{\gamma_k(x_{1:k}^i)}{\gamma_{k-1}(x_{1:k-1}^i) q_k(x_k^i | x_{1:k-1}^i)} \).

\[ \Rightarrow \{ x_{1:k}^i, w_k^i \}_{i=1}^N \]
SMC samplers are used to approximate a sequence of probability distributions on a sequence of probability spaces.

Using an artificial sequence of intermediate target distributions for an SMC sampler is a powerful (and quite possibly underutilised) idea.

**Key idea:** Perform and make use of various decompositions of graphical models to design SMC inference methods.
Outline

1. Example – from information theory
2. Sequential decomposition → “standard” SMC
   a) Sequential decomposition and SMC for PGMs
   b) Example – Estimating partition functions
3. Tree decomposition → *Divide-and-Conquer* with SMC
   a) Tree decomposition and D&C-SMC for PGMs
   b) Example – Hierarchical Bayesian Model

Fredrik Lindsten, *Sequential Monte Carlo for graphical models*
Statistics Seminar, University of Bristol, 7 November 2014.
Example borrowed from:


2D binary-input channel with the constraint that no two horizontally or vertically adjacent variables may be both be equal to 1.

```
... ... ... ... ...
... 0 1 0 ...
... 0 0 1 ...
... 0 1 0 ...
... ... ... ...
```

“of interest in magnetic and optical storage”

The channel can be described by a square lattice undirected graphical model.

Fredrik Lindsten, *Sequential Monte Carlo for graphical models*

Statistics Seminar, University of Bristol, 7 November 2014.
The variables are binary $x_{\ell,j} \in \{0, 1\}$ and the interactions are pair-wise between adjacent variables. Factors:

$$
\psi(x_{\ell,j}, x_{m,n}) = \begin{cases} 
0, & x_{\ell,j} = x_{m,n} = 1 \\
1, & \text{otherwise}
\end{cases}
$$
The resulting joint PDF is given by 

\[ p(X_\mathcal{V}) = \frac{1}{Z} \prod_{(\ell,j,m,n) \in \mathcal{E}} \psi(x_{\ell,j}, x_{m,n}) , \] 

For a channel of dimension \( M \times M \) we can write the finite-size noiseless capacity as 

\[ C_M = \frac{1}{M^2} \log_2 Z. \]

Unfortunately calculating \( Z \) exactly for these types of models is computationally prohibitive, since the complexity is exponential in the size of the grid \( M \).
Rewrite the PGM as a high-dimensional **undirected chain** by introducing a new set of variables $x_k$.

$$\phi(x_k) = \prod_{j=1}^{M-1} \psi(x_{j,k}, x_{j+1,k}),$$

$$\psi(x_{k-1}, x_k) = \prod_{j=1}^{M} \psi(x_{j,k-1}, x_{j,k}).$$
The undirected chain results in the following joint PDF

\[ p(X_V) = \frac{1}{Z} \prod_{k=1}^{M} \phi(x_k) \prod_{k=2}^{M} \psi(x_{k-1}, x_k). \]

Natural sequential decomposition:

\[ \gamma_k(x_{1:k}) = \prod_{\ell=1}^{k} \phi(x_{\ell}) \prod_{\ell=2}^{k} \psi(x_{\ell-1}, x_{\ell}). \]
For the 2D channel: **fully adapted** SMC sampler. To sample $x_k$ we run a forward/backward algorithm for the $k$th column.

This was just a special case, the important question is, can we do this for a general graphical model?! **Yes!**

Our SMC sampler compared to the **tree sampler** by


implemented according to

Key idea:

- Perform a **sequential decomposition** of the graphical model.
- Each **subgraph** induces an artificial target distribution.
- Apply SMC to the sequence of artificial target distributions.
The joint PDF of the set of random variables indexed by $\mathcal{V}$, $X_{\mathcal{V}} \triangleq \{x_1, \ldots, x_{|\mathcal{V}|}\}$

$$p(X_{\mathcal{V}}) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(X_C).$$

Example of a sequential decomposition of the above factor graph (the target distributions are built up by adding factors at each iteration),

\[ \gamma_1(X_{L_1}) \quad \gamma_2(X_{L_2}) \quad \gamma_3(X_{L_3}) \propto p(X_{\mathcal{V}}) \]
Let \( \{C_k\}_{k=1}^K \) be an ordered partition of \( C \). Define:

\[
\psi_k(X_{I_k}) \triangleq \prod_{C \in C_k} \psi_C(X_C),
\]

where \( I_k \subseteq \{1, \ldots, |\mathcal{V}|\} \) is the set of indices in the domain of \( \psi_k \).

The sequential decomposition is based on these factors,

\[
\gamma_k(X_{L_k}) \triangleq \prod_{\ell=1}^k \psi_\ell(X_{I_\ell}),
\]

where \( L_k \triangleq \bigcup_{\ell=1}^k I_\ell \).

By construction, \( L_K = \mathcal{V} \) and the joint PDF \( p(X_{L_K}) \propto \gamma_K(X_{L_K}) \).
Algorithm SMC sampler for graphical models

1. **Initialise** ($k = 1$): Draw $X^i_{L_1} \sim q_1(\cdot)$ and set $w^i_1 = W_1(X^i_{L_1})$.

2. **For** $k = 2$ to $K$ **do**:
   (a) Draw $a^i_k \sim \text{Cat} (\{w^i_{k-1}\}_{j=1}^N)$.
   (b) Draw $\xi^i_k \sim q_k(\cdot|X^{a^i_k}_{L_{k-1}})$ and set $X^i_{L_k} = X^{a^i_k}_{L_{k-1}} \cup \xi^i_k$.
   (c) Set $w^i_k = W_k(X^i_{L_k})$.

- Generates samples $\{X^i_{L_K}, w^i_K\}_{i=1}^N \text{ approx.} \sim p(X_{L_K})$.
- Provides an unbiased estimate of the **partition function**!
Example – Evaluation of topic models

Evaluating Latent Dirichlet Allocation models on heldout documents corresponds to estimating the partition function of a PGM.

\[ \alpha m \]
\[ \theta \]
\[ z_1 \cdots z_M \]
\[ w_1 \cdots w_M \]
\[ \Psi \]

(a) Synthetic

(b) PMC

(c) 20 newsgroups

Estimates of the log-likelihood of heldout documents for various datasets.

Fredrik Lindsten, *Sequential Monte Carlo for graphical models*
Statistics Seminar, University of Bristol, 7 November 2014.
Can be used for block sampling with PMCMC.

**Ex)** Iteratively update the white variables, conditionally on the black variables.

Potentially useful when no “natural” sequential decomposition is available for full graph.
The sequential decomposition is basically a chain-oriented decomposition of the PGM. This naturally leads to a sequence of distributions suitable for standard SMC samplers.

Divide-and-Conquer SMC:

**Key idea:**
- Consider graph decompositions organised on trees.
- Assign auxiliary target distributions to all nodes of the tree.
- Inference using a new class of SMC algorithms.
Hierarchical Bayesian network

We initialise the D&C-SMC with independent particle populations for each leaf in the tree decomposition. These are then merged, resampled and propagated as we move up the tree.

**Iter 1:** Initialise \((\tilde{x}^i_k, w^i_k)^N_{i=1}\) for \(k = 1, 2, 3\).

**Iter 2:** Merge populations 1 and 2 and propagate \(\Rightarrow (\tilde{x}^i_{1,2,4}, w^i_4)^N_{i=1}\)

**Iter 3:** Merge populations 3 and 4 and propagate \(\Rightarrow (\tilde{x}^i_{1,2,3,4,5}, w^i_5)^N_{i=1}\)
Tree decomposition follows naturally when the graphical model is a tree. However, the idea is more generally applicable.

Example: Lattice Markov random field

The subgraphs can be organised on a tree!
**Algorithm** \( \text{dc\_smc}(t) – \text{D&C-SMC for node } t \in T \)

1. For \( c \in C(t) \):
   1. \((x^i_c, w^i_c)_{i=1}^N \leftarrow \text{dc\_smc}(c)\).
   2. Resample \((x^i_c, w^i_c)_{i=1}^N\) to obtain the equally weighted particle system \((x^i_c, 1)_{i=1}^N\).

2. For particle \( i = 1, \ldots, N \):
   1. Simulate \( \tilde{x}^i_t \sim q_t(\cdot | \tilde{x}^i_{c_1}, \ldots, \tilde{x}^i_{c_C}) \) from some proposal kernel on \( \tilde{X}_t \), and where \((c_1, c_2, \ldots, c_C) = C(t)\).
   2. Set \( x^i_t = (\tilde{x}^i_{c_1}, \ldots, \tilde{x}^i_{c_C}, \tilde{x}^i_t) \).
   3. Compute \( w^i_t = \frac{\gamma_t(x^i_t)}{\prod_{c \in C(t)} \gamma_c(\tilde{x}^i_c)} \frac{1}{q_t(\tilde{x}^i_t | \tilde{x}^i_{c_1}, \ldots, \tilde{x}^i_{c_C})} \).

3. Return \((x^i_t, w^i_t)_{i=1}^N\).

- Generalises the SMC framework (std SMC recovered if \( T \) is a chain).
- Consistent and gives an unbiased estimate of the partition function.
D&C “Sampling Importance Resampling”

\[ \prod_{c \in C(t)} \hat{\gamma}_c^N(x_c) \rightarrow \text{Resampling} \rightarrow (\hat{x}_{c1}^i, \ldots, \hat{x}_{cC}^i)^{N}_{i=1} \rightarrow \text{Weighting} \rightarrow (x_t^i, w_t^i)^{N}_{i=1} \]
D&C-SMC extensions (II/III)

D&C-SMC: Auxiliary mixture sampling

\[ \prod_{c \in \mathcal{C}(t)} \hat{\gamma}_c^N(x_c) \rightarrow \text{(Auxiliary) Weighting} \rightarrow \text{Resampling} \]
D&C-SMC: Auxiliary mixture sampling + Tempering

\[ \prod_{c \in C(t)} \tilde{\gamma}_c^N (x_c) \rightarrow \text{(Auxiliary) Weighting} \rightarrow \text{Tempering} \]
Example – NY maths test (model)

Data  Table of test results (278 399 instances), with school code, year, number of students tested in that year and school, and the number students that passed.

Structure  We organise the data into a tree with the following form: NYC (root), borough of the school district, school district, school, year.

Parameters  
- Observations at the leaf (binomial $p_t = \text{logistic}(\theta_t)$).
- Parameters $\theta'_t = \theta_t + \Delta_e$, with $\Delta_e \sim \mathcal{N}(0, \sigma_e^2)$.
- Hyperparameters $\sigma_e^2 \sim \text{Exp}(1)$.

After marginalization of internal $\theta$-parameters, the dimensionality of the remaining parameters in the model is 3 555.
<table>
<thead>
<tr>
<th>Bronx</th>
<th>Queens</th>
</tr>
</thead>
<tbody>
<tr>
<td>10 Particles</td>
<td>100 Particles</td>
</tr>
<tr>
<td>-0.03(0.06)</td>
<td>-0.05(0.04)</td>
</tr>
<tr>
<td>0.07(0.07)</td>
<td>0.03(0.04)</td>
</tr>
</tbody>
</table>

Posterior distribution of $\delta_e =$”difference in logistic($\theta$) along edge $e$” for two boroughs (rows) and four computational regimes (columns), with mean and std dev below each histogram.
We compare our D&C-SMC (implemented in Java) to Hamiltonian Monte Carlo (Stan, implemented in C++).

Similar posterior approximation accuracy.

<table>
<thead>
<tr>
<th>Method</th>
<th>Iterations/Particles</th>
<th>Runtime</th>
</tr>
</thead>
<tbody>
<tr>
<td>D&amp;C-SMC</td>
<td>1000</td>
<td>39 s</td>
</tr>
<tr>
<td>HMC (Stan)</td>
<td>2000 (50% burn-in)</td>
<td>3860 s (64 min)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Node</th>
<th>Stan</th>
<th>D&amp;C-SMC</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manhattan</td>
<td>0.17</td>
<td>15.96</td>
<td>93.89</td>
</tr>
<tr>
<td>Bronx</td>
<td>0.05</td>
<td>8.12</td>
<td>165.69</td>
</tr>
<tr>
<td>Brooklyn</td>
<td>0.18</td>
<td>6.52</td>
<td>36.22</td>
</tr>
<tr>
<td>Queens</td>
<td>0.07</td>
<td>14.01</td>
<td>209.05</td>
</tr>
<tr>
<td>Staten Island</td>
<td>0.05</td>
<td>25.50</td>
<td>481.17</td>
</tr>
</tbody>
</table>

The effective samples per second and speedup.

Fredrik Lindsten, *Sequential Monte Carlo for graphical models*
Statistics Seminar, University of Bristol, 7 November 2014.
Conclusions

- We have derived SMC-based inference methods for graphical models of arbitrary topologies with discrete and/or continuous random variables.

- **Key insight:** We exploit various decompositions of the graphical model to design efficient SMC samplers.

- Examples involving:
  1. estimating the partition function
  2. inferring the latent variables
  3. learning parameters.

- If you have interesting and challenging problems involving graphical models, let us know!

Fredrik Lindsten, *Sequential Monte Carlo for graphical models*
Statistics Seminar, University of Bristol, 7 November 2014.
SMC (and PMCMC) methods for graphical models:


**Thank you!!**