Divide-and-Conquer with Sequential Monte Carlo

F. Lindsten, A. M. Johansen, Christian A. Naesseth, B. Kirkpatrick, T. B. Schön, J. Aston, A. Bouchard-Côté

1 The University of Cambridge, 2 The University of Warwick, Linköping University, University of Milan, Uppsala University, 4 The University of British Columbia

Summary

- Probabilistic Graphical Models (PGM) are routinely used in computer vision, localisation, error-correcting coding, speech recognition, computational biology, machine learning, etc.
- Inference in PGMs is in general intractable and approximate methods are needed.
- We propose to use Divide-and-Conquer SMC (D&C-SMC), a new SMC method generalizing the standard SMC (defined on chains) to a multipopulation and tree-oriented method.

Graphical Models

We consider models on the form,

\[ p(X_N) = \frac{1}{Z} \prod_{C \in C} \psi(x_C), \]

where the graph \( G = (V, E) \) has vertex set \( V = \{x_1, \ldots, x_N\} \), edge set \( E \), cliques \( C \) and \( Z = \int \prod_{C \in C} \psi(x_C) \, dx_N \) is the partition function (normalisation constant).

Even though the method applies to graphs of arbitrary topology, we will see that it is especially useful for tree-structured graphs such as hierarchical Bayesian models.

Divide-and-Conquer SMC

D&C-SMC is a method that utilises a divide-and-conquer strategy to approximate a distribution of interest. Multiple independent particle populations are resampled, merged and propagated as the method progresses on an auxiliary tree-structured decomposition (T) of the PGM.

Algorithm 1 dcsmc(t)

1. For \( c \in C(t) \),
   a. \( (x_i, w_i)^N_{i=1} \leftarrow \text{dcsmc}(c) \)
   b. Resample \( (x_i, w_i)^N_{i=1} \) to obtain the equally weighted particle system \( (x_i, 1)^N_{i=1} \).
2. For particle \( i = 1, \ldots, N \),
   a. Simulate \( x_i^t \sim q(x_i | x^{t-1}, \ldots, x^0) \) from some proposal kernel on \( X_t \), and where \( (c_1, c_2, \ldots, c_T) = C(t) \).
   b. Set \( x_i^t = (x_1^t, \ldots, x_T^t, x_{T+1}^t, \ldots, x_N^t) \).
   c. Set \( w_i^t = \prod_{C \in \mathcal{C}(t)} p(x_i^t) \prod_{i \in c} \gamma(x_i^t) \).
3. Compute \( Z^t = \left( \frac{1}{N} \sum_{i=1}^N w_i^t \right) \prod_{c \in \mathcal{C}(t)} Z^t_c \).
4. Return \( (x_i, w_i)^N_{i=1} \).

The above algorithm gives a recursive formulation of the basic D&C-SMC, extensions to mitigate degeneracy and improve computational efficiency are available. The procedure is consistent and gives an unbiased approximation of the normalisation constant \( Z \).

Hierarchical Bayesian Model - New York State Mathematics Test

In this example, we demonstrate the scalability of our method by analysing a dataset containing New York State Mathematics Test results for elementary and middle schools in New York City. The model contains, after Rao-Blackwellisation, 3,555 parameters that needs to be estimated.

Auxiliary tree decomposition

The auxiliary tree decomposition comes quite naturally when considering tree graphs, see example below. The disconnected components correspond to the groups of variables that are targeted by the different populations of the D&C-SMC algorithm.

Level 0:

Level 2:

Iteration 1 Simulate three independent particle systems \( \{x_i^t, w_i^t\}^N_{i=1}, j = 1, 2, 3 \).

Iteration 2 Resample particle population 1 and 2, merge these and sample \( \{x_i^t, w_i^t\}^N_{i=1}. \)

Iteration 3 Resample particle population 4 and 3, merge these and sample \( \{x_i^t, w_i^t\}^N_{i=1}. \)

We can also construct non-standard decompositions for arbitrary graphs as the one defined for a grid model with periodic boundary conditions in the figure below.

(b) The speedup compared to efficient implementation of Hamiltonian Monte Carlo (HMC) as implemented in STAN is shown above with effective samples per second.

Extensions

- Merging via mixture sampling
  Potentially improves approximation, at an increased computational cost.
- SMC samplers and tempering
  Use MCMC to rejuvenate particles and annealing between distributions.
- Particle MCMC
  D&C-SMC can be used within the PMCMC framework.
- Adaptation
  Adaptive resampling and population dependent particle system size.

Figure 2: Boxplots over 70 runs of each sampler, using \( N \) particles and a total of \( n \) annealing steps. D&C SMC (b) uses fewer particles, \( N = 50, 399 \) and 475 in the three columns (left to right), respectively, to match its computational cost to the other methods. Top: Estimates of \( \log(Z) \). The horizontal lines correspond to min and max of 10 runs of AIS with \( (N, n) = (10, 100000) \). Bottom: MSE (log-scale) for the estimates of \( E_{Z^t} \) (averaged over the grid).