Divide-and-Conquer sequential Monte Carlo for high dimensional filtering

Francesca R. Crucinio
Joint work with Adam M. Johansen

16 March 2023

BayesComp 2023

State Space Models (SSM)

$$\begin{tikzcd}
 & Y_{t-1} & Y_t & \\
\cdots \arrow[r] & X_{t-1} \arrow[u] \arrow[r, "f_t"] & X_t \arrow[u, "g_t"'] \arrow[r] & \cdots
\end{tikzcd}$$

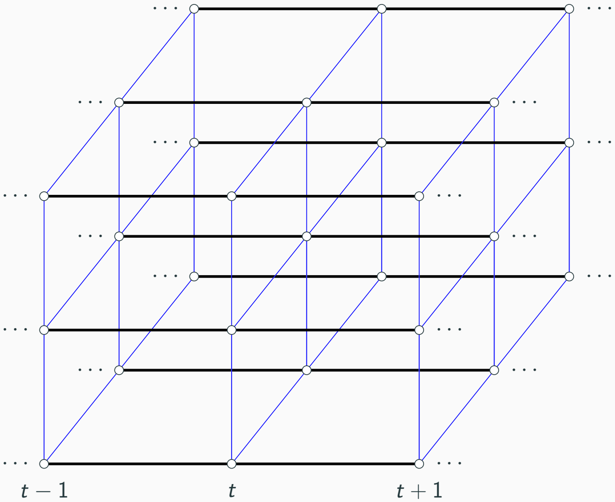
A state space model $(X_t, Y_t)_{t \geq 1} \subset \mathbb{R}^{d \times p}$ with

- transition density for the *latent* process $f_t(x_{t-1}, x_t)$
- likelihood for the *observation* process $g_t(x_t, y_t)$

Joint law is given by

$$\begin{aligned}
p(x_{1:t}, y_{1:t}) &= \prod_{k=1}^{t} f_k(x_{k-1}, x_k) g_k(x_k, y_k) \\
&= p(x_{1:t-1}, y_{1:t-1}) f_t(x_{t-1}, x_t) g_k(x_t, y_t),
\end{aligned}$$

Spatial SSM



Particle Filters (PF)

- **Aim:** Approximate

$$p(x_t|y_{1:t}) = g_t(x_t, y_t) \int f(x_{t-1}, x_t) p(x_{t-1}|y_{1:t-1}) \mathrm{d}x_{t-1}$$

using a set of N weighted particles $\{X_t^i, W_t^i\}_{i=1}^N$

- **Ingredients:**
  1. the transition density to sample $X_t^i \sim f_t(\tilde{X}_{t-1}^i, \cdot)$
  2. the likelihood to compute the weights $W_t^i \propto g_t(X_t^i, Y_t^i)$
  3. a resampling scheme to obtain $\{\tilde{X}_{t-1}^i, 1/N\}_{i=1}^N$ from $\{X_{t-1}^i, W_{t-1}^i\}_{i=1}^N$

Particle Filters for High-Dimensions

- Block PF (Rebeschini and Van Handel, 2015)
- Space-Time PF (STPF; Beskos et al. (2017))
- Nested sequential Monte Carlo (NSMC; Næsseth et al. (2015))

Divide and Conquer SMC for Filtering

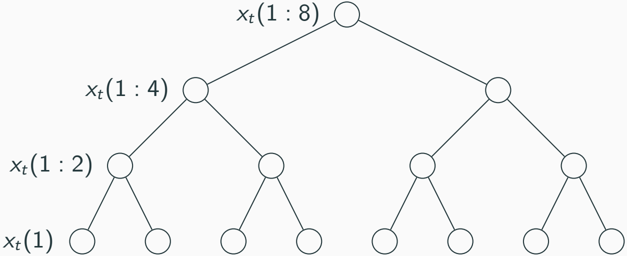


Figure 1: Space decomposition for $d = 8$.

Divide and Conquer SMC

An extension of standard SMC in which the sequence of target distributions $\{\gamma_u\}_{u \in \mathbb{T}}$ evolves on a tree $\mathbb{T}$ rather than on a line (Lindsten et al., 2017).

Need to define a sequence of distributions $\{\gamma_u\}_{u \in \mathbb{T}}$ so that at the root of the tree $\mathfrak{R}$ we have

$$\gamma_{\mathfrak{R}}(x_t(1:d)) = p(x_t | y_{1:t})$$

and for all other nodes γ_u only depends on a subset of $x_t(1:d)$, e.g. at the leaf level we have d distributions each depending on one component of x_t, $\gamma_u(x_t(i))$ for $i = 1, \ldots, d$.

Divide and Conquer SMC

Uses standard SMC ingredients

- transition/mutation
- reweighting
- resampling

with the addition of a **merging** step when two branches of the tree merge.

But instead of moving only “forward in time” we move “forward in space”.

The Merge Step

- **Aim:** Given two populations of weighted particles on the left and the right child, $\{X^i_{\ell(u)}, W^i_{\ell(u)}\}_{i=1}^N$ and $\{X^i_{r(u)}, W^i_{r(u)}\}_{i=1}^N$, approximating $\gamma_{\ell(u)}$ and $\gamma_{r(u)}$ build an approximation of γ_u

The Merge Step

- **Aim:** Given two populations of weighted particles on the left and the right child, $\{X^i_{\ell(u)}, W^i_{\ell(u)}\}_{i=1}^N$ and $\{X^i_{r(u)}, W^i_{r(u)}\}_{i=1}^N$, approximating $\gamma_{\ell(u)}$ and $\gamma_{r(u)}$ build an approximation of γ_u
- **Ingredients:**
  - the (weighted) product form estimator (Kuntz et al., 2022)

$$\gamma^N_{\ell(u)} \times \gamma^N_{r(u)} = \frac{1}{N^2} \sum_{i_1=1}^N \sum_{i_2=1}^N W^{i_1}_{\ell(u)} W^{i_2}_{r(u)} \delta_{(X^{i_1}_{\ell(u)}, X^{i_2}_{r(u)})}$$

  - N^2 *mixture* (importance) weights

$$m_u^{(i_2,i_2)} := W^{i_1}_{\ell(u)} W^{i_2}_{r(u)} \frac{\gamma_u(x^{i_1}_{\ell(u)}, x^{i_2}_{r(u)})}{\gamma_{\ell(u)}(x^{i_1}_{\ell(u)}) \gamma_{r(u)}(x^{i_2}_{r(u)})}$$

  - a resampling scheme to obtain $\{\tilde{X}^i_u, 1/N\}_{i=1}^N$ from $\{(X^{i_1}_{\ell(u)}, X^{i_2}_{r(u)}), m_u^{(i_2,i_2)}\}_{i_1,i_2=1}^N$

Choice of Distributions

Pick $f_{t,u}$ and $g_{t,u}$ proxies for marginals of f_t, g_t and define

$$\gamma_{t,u}(x_{t,u}) = g_{t,u}(x_{t,u}, (y_t(i))_{i\in u}) \int f_{t,u}(x_{t-1}, x_{t,u})\gamma_{t-1,\mathfrak{R}}(x_{t-1})\mathrm{d}x_{t-1}.$$

Choice of Distributions

Pick $f_{t,u}$ and $g_{t,u}$ proxies for marginals of f_t, g_t and define

$$\gamma_{t,u}(x_{t,u}) = g_{t,u}(x_{t,u}, (y_t(i))_{i \in u}) \int f_{t,u}(x_{t-1}, x_{t,u}) \gamma_{t-1,\mathfrak{R}}(x_{t-1}) \mathrm{d}x_{t-1}.$$

The integral above is intractable, so we define approximate targets as in marginal particle filters (Klaas et al., 2005)

$$\gamma_{t,u}(x_{t,u}) = g_{t,u}(x_{t,u}, (y_t(i))_{i \in u}) \frac{1}{N} \sum_{i=1}^{N} f_{t,u}(x^i_{t-1,\mathfrak{R}}, x_{t,u})$$

Pros and Cons

Pros

- No need for analytical form of $p(x_t(i) \mid x_t(1:i-1))$
- No need for factorized likelihoods
- Easy to parallelize and distribute

Cons

- Polynomial cost in N (can be mitigated via GPUs)
- Needs specification of γ_u

Linear Gaussian Model

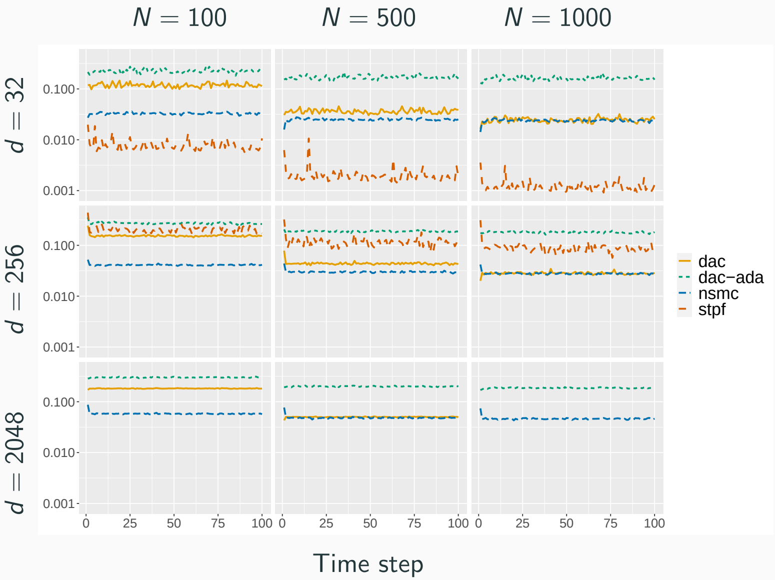
$$
\begin{aligned}
f_1(x_1) &= \mathcal{N}(x_1; m_1, \Sigma_1) \\
f_t(x_{t-1}, x_t) &= \mathcal{N}(x_t; Ax_{t-1}, \Sigma) \\
g_t(x_t, y_t) &= \mathcal{N}(y_t; x_t, \sigma_y^2 Id_d),
\end{aligned}
$$

with $\Sigma \in \mathbb{R}^{d \times d}$ a tridiagonal matrix.

We compare the RMSE

$$
\mathrm{RMSE}(x_t(i)) := \frac{\mathbb{E}\left[(\bar{x}_t(i) - \mu_{t,i})^2\right]}{\sigma_{t,i}^2}
$$

Linear Gaussian Model - RMSE



Spatial Model

The components of X_t are indexed by the vertices $v \in V$ of a lattice, where $V = \{1, \ldots, d\}^2$, and

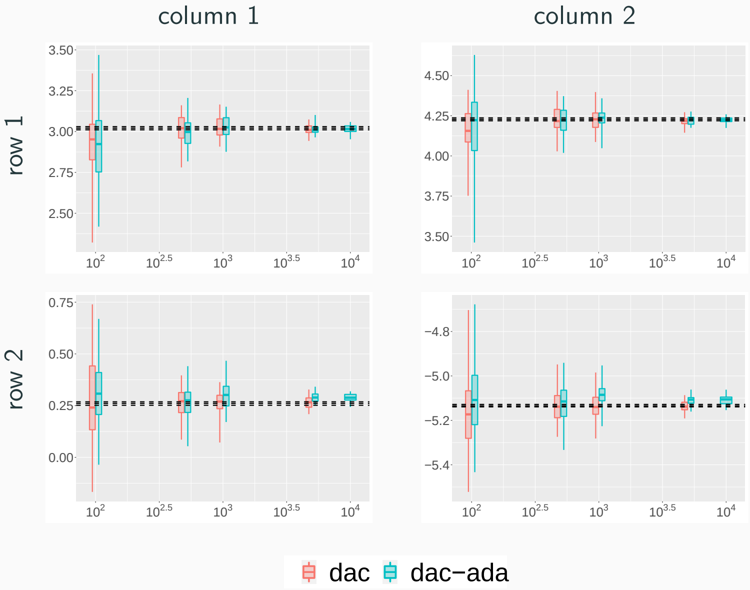
$$X_t(v) = X_{t-1}(v) + U_t(v), \qquad U_t(v) \sim \mathcal{N}(0, \sigma_x^2)$$

and

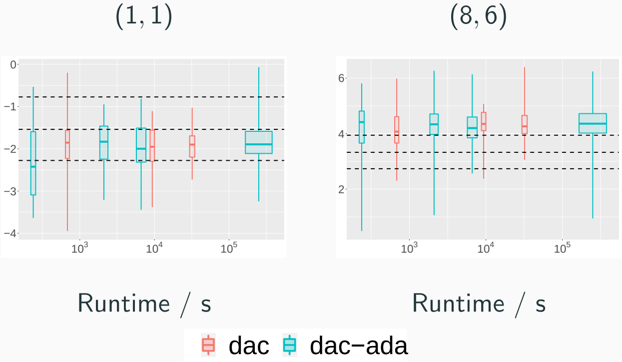
$$g_t(x_t, y_t) \propto$$

$$\left[1 + \nu^{-1} \sum_{v \in V} \left((y_t(v) - x_t(v)) \sum_{j : D(v,j) \leq r_y} \tau^{D(v,j)} (y_t(j) - x_t(j)) \right) \right]^{-(\nu + d^2)/2}.$$

2×2 grid



8×8 grid



- Theoretical guarantees should follow from those of standard DaC-SMC (Kuntz et al., 2021) and marginal particle filters (Crucinio and Johansen, 2023)
- Test on real models/data
- smoothing, parameter estimation, ...
- lookahead methods

Thank you!

References

Alexandros Beskos, Dan Crisan, Ajay Jasra, Kengo Kamatani, and Yan Zhou. A stable particle filter for a class of high-dimensional state-space models. *Advances in Applied Probability*, 49(1):24–48, 2017.

N Chopin and O Papaspiliopoulos. *An Introduction to Sequential Monte Carlo*. Springer, Cham, 2020. doi: 10.1007/978-3-030-47845-2.

F. R. Crucinio and A. M. Johansen. Properties of marginal sequential Monte Carlo methods. *arXiv preprint arXiv:2303.03498*, 2023.

Mike Klaas, Nando De Freitas, and Arnaud Doucet. Toward practical N^2 Monte Carlo: The marginal particle filter. In *Proceedings of the 21st Conference in Uncertainty in Artificial Intelligence*, pages 308–315, 2005.

J. Kuntz, F. R. Crucinio, and A. M. Johansen. The divide-and-conquer sequential Monte Carlo algorithm: theoretical properties and limit theorems. *arXiv preprint arXiv:2110.15782*, 2021.

Juan Kuntz, Francesca R Crucinio, and Adam M Johansen. Product-form estimators: exploiting independence to scale up Monte Carlo. *Statistics and Computing*, 32(1):1–22, 2022.

Fredrik Lindsten, Adam M Johansen, Christian A Næsseth, Bonnie Kirkpatrick, Thomas B Schön, John A D Aston, and Alexandre Bouchard-Côté. Divide-and-Conquer with sequential Monte Carlo. *Journal of Computational and Graphical Statistics*, 26(2):445–458, 2017.

Christian A Næsseth, Fredrik Lindsten, and Thomas B Schön. Nested sequential Monte Carlo methods. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, pages 1292–1301. Proceedings of Machine Learning Research, 2015.

Patrick Rebeschini and Ramon Van Handel. Can local particle filters beat the curse of dimensionality? *Annals of Applied Probability*, 25 (5):2809–2866, 2015.